A PARTICLE FILTERING-BASED FRAMEWORK FOR ON-LINE FAULT DIAGNOSIS AND FAILURE PROGNOSIS

A Thesis
Presented to
The Academic Faculty

By

Marcos E. Orchard

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in
Electrical and Computer Engineering

Georgia Institute of Technology

December 2007

A PARTICLE FILTERING-BASED FRAMEWORK FOR ON-LINE FAULT DIAGNOSIS AND FAILURE PROGNOSIS

Approved by:

Dr. George J. Vachtsevanos, Advisor
School of Electrical and Computer Engineering

Dr. Aldo A. Ferri
School of Mechanical Engineering

Dr. Ronald G. Harley
School of Electrical and Computer Engineering

Dr. James H. McClellan
School of Electrical and Computer Engineering

Dr. Jennifer E. Michaels
School of Electrical and Computer Engineering

Date Approved: November 02, 2007

To my family, heart and soul of my existence,

and to all those who have shown me love and kindness throughout life.

Truly yours, Marcos.

"All we have to decide is what to do with the time that is given to us"

J.R.R. Tolkien*, The Fellowship of the Ring, 1954.*

ACKNOWLEDGMENTS

The author wishes to express infinite gratitude to Dr. George Vachtsevanos, advisor of this research work, for his precious friendship, guidance, and support throughout all these years. His dedication and example have had a tremendous impact in the author, as much as a person as a researcher, and will be always remembered. Dear Dr. Vachtsevanos, for each one of those "buenos días?" and "cómo está?", thank you so much.

Appreciation is also expressed to the dissertation committee members, Dr. Aldo Ferri, Dr. Ronald G. Harley, Dr. James H. McClellan, and Dr. Jennifer Michaels. Their comments, suggestions, and corrections contributed enormously to improve the quality of this work, as well as the presentation of the material.

Support for completing the doctoral degree was provided in part by the President of the Republic Scholarship (Chile), Fulbright-Chile, MECESUP-Chile, and the University of Chile, and is deeply appreciated.

The author also wants to show his appreciation to all the members and friends of the Intelligent Control Systems Laboratory of Georgia Tech, not only for the endless (and very entertaining) conversations, but also for their support and genuine care. Thanks to each and every one of you: Mr. Gary O'Neill, Dr. Biqing Wu, Dr. Otis Smart, Bhaskar, Doug, Jackie, Matt, Manzar, Mourlas, Lauren, Taimoor, and Ms. Sharon Lawrence. Particular mention and gratitude is expressed to Dr. Romano Patrick ("brother in

research"), Dr. Abhinav Saxena, Dr. George Georgoulas (nonstop supplier of movies and Greek sweets), and Dr. Bin Zhang. The author also wishes to specially thank the friendship and comradeship of Claudio, Harjeet and Jean Carlos: the moments shared and enjoyed here at Georgia Tech will last a lifetime.

It is the desire of the author to express special gratefulness to his middle-school, high-school, and college friends: although life may have taken us through different paths, your kindness and generosity has never been forgotten. In special, the author wants to thank his high-school teachers from "Colegio San Luis", some of them among the first people who ever addressed him as "Dr. Orchard": Father Cristian Brahm, Otto, Nancy, Beto, Luco, "Charro" Martínez, Gladys, Tirsa, and many others. Special thanks are conveyed to Dr. Aldo Cipriano and Dr. Juan Dixon, from Pontificia Universidad Católica de Chile, for their friendship, support and guidance during undergraduate studies.

There are not enough words to convey the immense love and most sincere gratefulness that the author desires to express to his family and his wife, Yamille. They are the ones that really made it possible to be here and to fulfill this dream. Throughout hardships they always lent a helping hand, and in happier times they always shared the joy with me. Your sacrifices, your support, your jokes and laughs, your hugs and tenderness, and also your tears: these are the substance and the soul of my being, my heart, and my life. Truthfully, thank you!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ACM	Automated contingency management
ASIR	Auxiliary particle filter
EKF	Extended Kalman filter
FD	Fault detection
FDI	Fault detection and isolation
FEA	Finite element analysis
FMECA	Failure modes, effects, and criticality analysis
GAG	Ground-air-ground
GUI	Graphical user interface
HPC	High-power compressor
IBS	Inter-blade spacing
i.i.d.	Independent and identically distributed
LLR	Logarithms of the likelihood ratio
MCMC	Markov chain Monte Carlo
MH	Metropolis-Hastings
MLE	Maximum likelihood estimation
pdf	Probability density function
PF	Particle filtering
RBPF	Rao-Blackwellised particle filter
RPF	Regularized particle filter
RSPF	Risk-sensitive particle filter
RUL	Remaining useful life

SBR	Sideband ratio
SIS	Sequential importance sampling
SISR	Sequential importance sampling resampling
SMC	Sequential Monte Carlo
SS	State estimation
TBP	Tangential blade position
TOA	Time-of-arrival
TTF	Time-to-failure
UPF	Unscented particle filter
UKF	Unscented Kalman filter
VRPF	Variable resolution particle filter
VUF	Variable resolution unscented filter
WSS	Wide sense stationary

SUMMARY

This thesis presents an on-line particle-filtering-based framework for fault diagnosis and failure prognosis in nonlinear, non-Gaussian systems. The methodology assumes the definition of a set of fault indicators, which are appropriate for monitoring purposes, the availability of real-time process measurements, and the existence of empirical knowledge (or historical data) to characterize both nominal and abnormal operating conditions.

The incorporation of particle-filtering (PF) techniques in the proposed scheme not only allows for the implementation of real time algorithms, but also provides a solid theoretical framework to handle the problem of fault detection and isolation (FDI), fault identification, and failure prognosis. Founded on the concept of sequential importance sampling (SIS) and Bayesian theory, PF approximates the conditional state probability distribution by a swarm of points called "particles" and a set of weights representing discrete probability masses. Particles can be easily generated and recursively updated in real time, given a nonlinear process dynamic model and a measurement model that relates the states of the system with the observed fault indicators.

Two autonomous modules have been considered in this research. On one hand, the fault diagnosis module uses a hybrid state-space model of the plant and a particle-filtering algorithm to (1) calculate the probability of any given fault condition in real time, (2) estimate the probability density function (pdf) of the continuous-valued states in the monitored system, and (3) provide information about *type I* and *type II* detection

errors, as well as other critical statistics. Among the advantages offered by this diagnosis approach is the fact that the pdf state estimate may be used as the initial condition in prognostic modules after a particular fault mode is isolated, hence allowing swift transitions between FDI and prognostic routines.

The failure prognosis module, on the other hand, computes (in real time) the pdf of the remaining useful life (RUL) of the faulty subsystem using a particle-filtering-based algorithm. This algorithm consecutively updates the current state estimate for a nonlinear state-space model (with unknown time-varying parameters) and predicts the evolution in time of the fault indicator pdf. The outcome of the prognosis module provides information about the precision and accuracy of long-term predictions, RUL expectations, 95% confidence intervals, and other hypothesis tests for the failure condition under study. Finally, inner and outer correction loops (learning schemes) are used to periodically improve the parameters that characterize the performance of FDI and/or prognosis algorithms. Illustrative theoretical examples and data from a seeded fault test for a UH-60 planetary carrier plate are used to validate all proposed approaches.

Contributions of this research include: (1) the establishment of a general methodology for real time FDI and failure prognosis in nonlinear processes with unknown model parameters, (2) the definition of appropriate procedures to generate dependable statistics about fault conditions, and (3) a description of specific ways to utilize information from real time measurements to improve the precision and accuracy of the predictions for the state probability density function (pdf).

1. INTRODUCTION

1.1. Overview and Objective of this Thesis

Fault diagnosis and failure prognosis in complex systems have become key issues in a world where the economic impact of system reliability and cost-effective operation of critical assets is steadily increasing. On the one hand, failure diagnosis involves the detection of a fault in the system, its isolation, and the assessment of its severity. On the other hand, prognosis – as a natural extension to the fault detection and isolation (FDI) problem – intends to characterize the evolution in time of the incipient failure condition, thus allowing the estimation of the remaining useful life (RUL) for affected subsystems or components.

Several examples can be cited here to illustrate the range of applications for these types of algorithms: electro-mechanical systems, continuous-time manufacturing processes, structural damage analysis, and even fault tolerant software architectures. Most of them have in common the fact that they are highly complex, nonlinear, and affected by large-grain uncertainty.

The task can be particularly difficult when the system under study is operating in real-time, especially when prognostic algorithms are implemented. Most of the approaches currently available in the reliability arena require intensive computations and the processing of large amounts of historical data. More importantly, the obtained results do not necessarily include knowledge about the physics of the system and there is little

room left for on-line updates in the predicted RUL when the system is behaving differently from what is expected. Learning paradigms ─ so useful in the control field ─ are rarely applied for prognostic purposes, thus limiting the implementation of automatic contingency management (ACM) systems or other automated corrective schemes. In addition, even in the case when efficient FDI algorithms are implemented, there are no unified approaches that can perform the transition from FDI results to prognostic modules.

The objective of this thesis is to establish a general framework to deal with the problems of real-time fault diagnosis and failure prognosis via the utilization of particle-filtering (PF) techniques, an emerging methodology for sequential signal processing that is very suitable when the system is nonlinear or in the presence of non-Gaussian process/observation noise. The present work achieves this goal by simultaneously accomplishing three specific objectives.

The first specific objective is to implement an on-line particle-filtering-based framework for fault diagnosis in nonlinear, non-Gaussian systems. This architecture must be able to pinpoint both the presence and nature of a fault condition in real time, given a set of measurements and a characterization of the plant behavior under nominal operational conditions (baseline data). Furthermore, this scheme provides the means for a rapid transition between the FDI and prognostic applications.

The second specific objective of this thesis is to use a particle-filtering-based framework for on-line failure prognosis in nonlinear, non-Gaussian systems. This

implementation statistically characterizes the remaining useful life (RUL) of a subsystem or piece of equipment affected by a fault condition, i.e., estimates the probability density function of the subsystem RUL. A set of measurements is used to improve current estimates, and nonlinear state-space models with unknown time varying parameters define the evolution in time of the fault indicator. The outcome of the prognosis module, namely the RUL pdf, is available and is updated in real time, providing information about statistical confidence intervals, expectations, and other hypothesis tests for the failure condition under study.

The third, and last, objective for this thesis is to establish learning schemes whereby the information acquired on-line, from measurements, is transformed into a set of corrections for the parameters that characterize the performance of FDI and/or prognosis algorithms. According to the nature of the parameters that these schemes modify, these loops are classified into two different categories: *inner* and *outer correction loops*.

Data from a seeded fault test for a UH-60 planetary carrier plate are used to validate all proposed approaches. Other academic and illustrative examples are also implemented to illustrate the advantages and disadvantages of the proposed methodology with respect to the current state-of-the-art in the field.

1.2. Principal Contributions

This thesis presents several approaches that can be used to implement a real-time particle-filtering-based framework for fault diagnosis and failure prognosis. These approaches share the premise that the current state pdf estimate can be used to determine the operating condition of the system and/or to predict the progression of a fault indicator, given a dynamic state model and process measurements. In particular, the principal contributions of this work are:

- A general framework for fault diagnosis. Results indicate that the proposed particle-filtering-based methodology is successful and very efficient in pinpointing abnormal conditions in real time for a variety of cases, given the definition of a hybrid nonlinear state-space model for the system, a characterization of the plant behavior under nominal operational conditions (baseline data), and the availability of real-time measurements. This framework also allows performing rapid transitions between FDI and prognostic-oriented applications.

- Methodologies to estimate the probability of a fault in real time. These methodologies include a step-by-step procedure to compute *type I* and *type II* detection errors, and also the means to execute classical statistical hypothesis tests in nonlinear systems and in the presence of non-Gaussian noise. These methodologies also provide means for detecting the existence of unknown anomalies in the monitored system with a specified confidence level.

- A novel framework for failure prognosis that is capable of estimating the probability of failure at future time instants (RUL pdf) in real time. This methodology combines state pdf estimates, long-term predictions, and empirical knowledge about critical conditions for the system (also referred to as the hazard zones) to provide information about time-to-failure (TTF) expectations, statistical confidence intervals, and other hypothesis tests. Particularly, it is shown that a combination of resampling schemes in long-term predictions and Epanechnikov kernels helps to reduce the impact of model errors and simultaneously offers a balanced answer in terms of accuracy and precision in RUL estimates. In addition, it is shown that an approach based solely on the expectation of the long-term prediction also provides acceptable results and, moreover, it is very suitable for on-line applications with limited computational resources.

- The implementation of *outer correction loops* to update parameters of great significance in the overall performance of FDI and/or prognosis algorithms. These *correction loops* are (1) an autoregressive correction algorithm utilized to improve accuracy in RUL expectations, and (2) a model parameter update procedure that facilitates identification of nonlinear systems undergoing changes in operational conditions. Both illustrate how the accuracy of the prognosis algorithm may be significantly enhanced when several learning loops – combining model-based and data driven techniques – are working in parallel.

- The validation of the proposed framework in three case studies, using simulated and real failure data. These studies provide excellent insight about how model

inaccuracies and/or customer specifications (hazard zone definition or desired prediction window) affect the algorithm performance.

1.3. Organization of the thesis

There are four major parts to this thesis. The first part, contained in Chapter 2, provides a general overview of the nonlinear Bayesian estimation problem, focusing on the principal aspects of sequential Monte Carlo algorithms (also referred to as particle-filtering algorithms) and the most important variants present in the literature. In addition, this chapter gives a brief account of the state-of-the-art in the application of these techniques in the fields of fault diagnosis and failure prognosis.

The second part, contained in Chapter 3, presents the implementation of a particle-filter-based framework for on-line fault diagnosis. In particular, the chapter starts with a general description of the framework, and follows with two specific application examples to illustrate the details of the implementation issues. Additionally, a novel approach for an anomaly detector is introduced at the end of this chapter.

The third part of this thesis is comprised of Chapters 4 and 5, where a particle-filter-based framework for on-line failure prognosis is presented. Chapter 4 introduces the theoretical aspects of the proposed methodology, while Chapter 5 focuses on the description and analysis of three case studies. The last part of the thesis, Chapter 6, states the conclusions and recommended future work.

2. HISTORICAL BACKGROUND

2.1. Scope and Objective of the Chapter

The performance and efficiency of any model-based approach for fault diagnosis and failure prognosis will rely, to a great extent, on the ability of the dynamic model to mimic the behavior of the process under study. Linear and Gaussian dynamic models may help to describe this behavior satisfactorily when either the process complexity is reduced or when the time framework intended for long-term predictions is shortened. Most of the time, though, real processes require the inclusion of nonlinear dynamics or non-Gaussian stochastic components for an accurate description, especially when the time horizon required for the generation of dependable prognosis results is long enough to make evident any deficiencies/shortcomings introduced through linearization methodologies.

Nonlinear Bayesian and sequential Monte Carlo (SMC) methods provide a solid and consistent theoretical framework to handle the state estimation problem [1] under the conditions mentioned above, and thus they offer the means to implement both diagnostic and prognostic algorithms. Both methods have been the subject of a broad and intensive amount of research over the past years in many diverse disciplines, including economics, biostatistics, and even statistical signal processing problems in the engineering domain such as time series analysis, radar and sonar target tracking, and communications [2].

In this sense, before highlighting the contributions that these techniques may offer in the area of FDI and prognostics, a comprehensive theoretical review is presented on the state-of-the-art in SMC methods, also referred to as particle filters, with particular emphasis on state-space (SS) estimation and model parameter identification.

With the intention of achieving this objective, the present chapter is structured as follows. Section 2.2 presents both the definition and the general formulation for the nonlinear Bayesian filtering problem, as well as an introductory notion about how SMC methods can help to actually solve this problem. Section 2.3 focuses on theoretical aspects behind the implementation of SMC methods, and in particular of the sequential importance sampling resampling (SISR) particle filter, including the main limitations for these types of algorithms. In addition, section 2.3 presents a summary of the most recent improvements in the field, with emphasis on those that can be useful for applications related to FDI and/or model parameter estimation.

Finally, sections 2.4 and 2.5 provide a general overview of the state of the art for the application of particle filtering algorithms in the field of real-time diagnosis and prognosis. These methodologies and published results – in areas such as robotics, automation, and artificial intelligence – are the foundation for the novel approaches presented in this thesis.

2.2. Nonlinear Bayesian Filtering

Nonlinear filtering is defined as the process of using noisy observation data to estimate at least the first two moments of a state vector governed by a dynamic nonlinear, non-Gaussian state-space model [3]. Although in principle the estimation procedure may be implemented on continuous-time systems, the present research is solely focused on discrete-time systems since the streaming measurement data is sent and received through digital devices in most of the applications relevant to FDI and prognosis.

Within a Bayesian general formulation for the dynamic state estimation problem, a nonlinear filtering procedure intends to generate an estimate of the posterior probability density function (pdf) for the state, based on the set of received measurements [3]. Since such an estimate for the state vector is required almost every time measurement data are received, it makes sense to use a recursive strategy to update the estimation results. Such strategy avoids the problem of massive data storage and/or recalculation of the whole state trajectory in time.

Mathematically, let an unobserved process $X = \{X_t, t \in \mathbb{N}\}$ be an $\mathbb{R}^{n_x}-$valued Markov process characterized both by its initial distribution $p(x_0)$ and the transition probability $p(x_t \mid x_{t-1})$. Let $p(x_t \mid x_{t-1})$ be defined by state equation (2.01), where $\{\omega_t\}_{t\geq 0}$ is a sequence of independent random variables.

$$x_t = f_t(x_{t-1}, \omega_t) \tag{2.01}$$

Noisy observations $Y = \{Y_t, t \in \mathbb{N}\}$ are accessible and assumed to be conditionally independent given the process $X = \{X_t, t \in \mathbb{N}\}$. Equation (2.02) defines the distribution of $Y_t \mid X_t$ and, hence, of the marginal distribution $p(y_t \mid x_t)$, where $\{\upsilon_t\}_{t \geq 0}$ is a sequence of independent random variables.

$$y_t = h_t(x_t, \upsilon_t) \tag{2.02}$$

Let $x_{0:t} \triangleq \{x_0, \cdots, x_t\}$ and $y_{1:t} \triangleq \{y_1, \cdots, y_t\}$ denote, respectively, the signal and the observations up to time *t*. It is of interest to estimate the *posterior distribution* $p(x_{0:t} \mid y_{1:t})$, the marginal distribution $p(x_t \mid y_{1:t})$ (also referred to as the *filtering distribution*) and the expectations

$$I(f_t) = \mathrm{E}_{p(x_{0:t} \mid y_{1:t})}[g_t(x_{0:t})] \triangleq \int g_t(x_{0:t}) p(x_{0:t} \mid y_{1:t}) dx_{0:t}\,, \tag{2.03}$$

for any function $g_t : \mathbb{R}^{n_x} \to \mathbb{R}^{n_{f_t}}$ integrable with respect to $p(x_{0:t} \mid y_{1:t})$, such as the conditional mean of x_t (1^{st} moment, where $g_t(x_{0:t}) = x_{0:t}$) or the conditional variance of x_t (2^{nd} moment, where $g_t(x_{0:t}) = x_t x_t^{\,T} - E_{p(x_t \mid y_{1:t})}[x_t] E^T_{p(x_t \mid y_{1:t})}[x_t]$) [2].

This task can be achieved by performing two sequential steps, namely *prediction* and *filtering* [1]. On the one hand, *prediction* uses both the knowledge of the previous state estimate and the process model to generate the *a priori* state pdf estimate for the next time instant, as shown in (2.04).

$$p(x_{0:t} \mid y_{1:t-1}) = \int p(x_t \mid x_{t-1}) p(x_{0:t-1} \mid y_{1:t-1})\, dx_{0:t-1} \tag{2.04}$$

On the other hand, the *filtering step* (2.05) considers the current observation y_t, the *a priori* state pdf, the *likelihood* function $p(y_t \mid x_t)$, and Bayes formula to generate the *posterior* state pdf $p(x_{0:t} \mid y_{1:t})$.

$$\begin{aligned} p(x_{0:t} \mid y_{1:t}) &= \frac{p(y_{1:t} \mid x_{0:t}) p(x_{0:t})}{p(y_{1:t})} = \frac{p(y_t, y_{1:t-1} \mid x_{0:t}) p(x_{0:t})}{p(y_t, y_{1:t-1})} \\ &= \frac{p(y_t \mid x_{0:t}, y_{1:t-1}) p(y_{1:t-1} \mid x_{0:t}) p(x_{0:t})}{p(y_t \mid y_{1:t-1}) p(y_{1:t-1})} \\ &= \frac{p(y_t \mid x_{0:t}, y_{1:t-1}) p(x_{0:t} \mid y_{1:t-1}) p(y_{1:t-1}) p(x_{0:t})}{p(y_t \mid y_{1:t-1}) p(y_{1:t-1}) p(x_{0:t})} \\ &= \frac{p(y_t \mid x_t) p(x_{0:t} \mid y_{1:t-1})}{p(y_t \mid y_{1:t-1})} \end{aligned} \tag{2.05}$$

Furthermore, as (2.06) shows, the *filtering* step may be implemented by using the recursion formula:

$$\begin{aligned} p(x_{0:t} \mid y_{1:t}) &= \frac{p(y_t \mid x_t) p(x_{0:t} \mid y_{1:t-1})}{p(y_t \mid y_{1:t-1})} \\ &= \frac{p(y_t \mid x_t) p(x_t \mid x_{0:t-1}, y_{1:t-1})}{p(y_t \mid y_{1:t-1})} \cdot p(x_{0:t-1} \mid y_{1:t-1}), \\ &= \frac{p(y_t \mid x_t) p(x_t \mid x_{0:t-1})}{p(y_t \mid y_{1:t-1})} \cdot p(x_{0:t-1} \mid y_{1:t-1}) \end{aligned} \tag{2.06}$$

where $p(y_t \mid y_{1:t-1}) = \int p(y_t \mid x_t) p(x_t \mid y_{1:t-1})\, dx_t$.

The recursive computation of the *posterior* state pdf $p(x_{0:t} \mid y_{1:t})$ is more conceptual than practical, however, since the integrals in (2.04) and (2.06) do not have an analytical solution in most cases [1], [4]. Similarly, the computation of any expectation, as in (2.03), involves the solution of an integration process, which usually does not have a closed-form solution.

For aforementioned reason, since the mid-1960s, numerous investigators have attempted to arrive at approximations that could minimize the variance of these integral estimates. Some examples of these methodologies are [1]-[2]: the Extended Kalman Filter, the Gaussian sum filter, and grid-based methods. In particular, it is important to note that grid-based methods actually provide the optimal estimate for the filtering distribution when the state-space is discrete and consists of a finite number of states [1]. Numerical methods became increasingly interesting for the scientific community in the late 1980s, when the accelerated increase in computational capabilities made possible to compute and obtain results from Monte Carlo-based algorithms in a reasonable amount of time, especially if efficient sampling methods are used, such in the case of SMC (particle filters), which are now explained in detail.

2.3. Sequential Monte Carlo Methods: Particle Filtering

Consider a sequence of probability distributions $\{\pi_t(x_{0:t})\}_{t\geq 1}$, where it is assumed that $\pi_t(x_{0:t})$ can be evaluated pointwise up to a normalizing constant. SMC methods, also referred to as Particle Filters (PF), are a class of algorithms designed to approximately

obtain samples sequentially from $\{\pi_t\}$, i.e., to generate a collection of $N >> 1$ weighted random samples $\{w_t^{(i)}, x_{0:t}^{(i)}\}_{i=1\cdots N}$, $w_t^{(i)} \geq 0, \forall t \geq 1$, satisfying (2.07) [5]:

$$\sum_{i=1}^{N} w_t^{(i)} \varphi_t(x_{0:t}^{(i)}) \xrightarrow[N\to\infty]{} \int \varphi_t(x_{0:t}) \pi_t(x_{0:t}) dx_{0:t}, \tag{2.07}$$

where φ_t is any π_t – integrable function.

In the particular case of the Bayesian filtering problem, the *target distribution* $\pi_t(x_{0:t}) = p(x_{0:t} \mid y_{1:t})$ is the *posterior* pdf of $X_{0:t}$, given a realization of noisy observations $Y_{1:t} = y_{1:t}$. Furthermore, from (2.01) and (2.02), the *posterior* may be written as [4]

$$\pi_t(x_{0:t}) = p(x_0) \prod_{k=1}^{t} f_k(x_k \mid x_{k-1}) h_k(y_k \mid x_k). \tag{2.08}$$

The most basic SMC algorithm, the SIR particle filter (a.k.a. bootstrap filter), solves the problem stated in (2.07) using a s*equential importance sampling resampling* (SISR) scheme. The following subsections present a description of this scheme.

2.3.1. Importance Sampling Scheme

Assume that a set of *N* paths $\{x_{0:t-1}^{(i)}\}_{i=1\cdots N}$, distributed according to $\pi_{t-1}(x_{0:t-1})$, is available at time $t-1$, i.e., it is possible to approximate $\pi_{t-1}(x_{0:t-1})$ with the empirical distribution

$$\pi_{t-1}^{N}(x_{0:t-1}) = \frac{1}{N}\sum_{i=1}^{N}\delta(x_{0:t-1} - x_{0:t-1}^{(i)}). \tag{2.09}$$

The objective is to efficiently obtain a set of N new paths (particles) $\{\tilde{x}_{0:t}^{(i)}\}_{i=1\cdots N}$ that are approximately distributed according to $\pi_t(\tilde{x}_{0:t})$. To generate this new set of particles, all current paths $x_{0:t-1}^{(i)}$ are extended using the kernel function $q_t(\tilde{x}_{0:t} \mid x_{0:t-1})$. Although it is desired for these newly generated paths to distribute as close as possible to $\pi_t(\tilde{x}_{0:t})$, their actual distribution is given by (2.10) [5].

$$q_t(\tilde{x}_{0:t}) = \int_{X^t} q_t(\tilde{x}_{0:t} \mid x_{0:t-1})\,\pi_{t-1}(x_{0:t-1})dx_{0:t-1} \tag{2.10}$$

Thus, to generate consistent estimates for (2.03), it is necessary to correct for the differences that, in practice, will exist between the distributions $q_t(\tilde{x}_{0:t})$ (also referred to as the *importance* density function) and $\pi_t(\tilde{x}_{0:t})$. Importance sampling deals with this problem by setting the values of the *N* weights $\{w_t^{(i)}\}_{i=1\cdots N}$ equal to the ratio (Radon-Nikodym derivative):

$$w(\tilde{x}_{0:t}) = \frac{\pi_t(\tilde{x}_{0:t})}{q_t(\tilde{x}_{0:t})}. \tag{2.11}$$

Following this reasoning, and considering the importance sampling identity (2.12), it is possible to prove that (2.13) represents a Monte Carlo approximation that converges to the desired distribution $\pi_t(x_{0:t})$ [5]:

$$\pi_t(\tilde{x}_{0:t}) = \frac{w(\tilde{x}_{0:t})\, q_t(\tilde{x}_{0:t})}{\int_{X^{t+1}} w(\tilde{x}_{0:t})\, q_t(\tilde{x}_{0:t})\, d\tilde{x}_{0:t}} \tag{2.12}$$

$$\tilde{\pi}_t^N(x_{0:t}) = \sum_{i=1}^{N} w_{0:t}^{(i)} \delta(x_{0:t} - \tilde{x}_{0:t}^{(i)}), \tag{2.13}$$

where $w_{0:t}^{(i)} \propto w_{0:t}(\tilde{x}_{0:t}^{(i)})$ and $\sum_{i=1}^{N} w_{0:t}^{(i)} = 1$.

The efficiency of this procedure improves as the variance of the importance weights is minimized, which basically means that the importance distribution is close to the actual target distribution. Although the overall procedure seems to be simple, it requires evaluating (2.10) in a closed form, or equivalently, evaluating $q_t(\tilde{x}_{0:t})$ up to a normalizing constant, which usually is impossible. Nevertheless, there exists a significant case where interesting conclusions can be attained.

In fact, consider the case when the importance distribution function is of the form (2.14), which is equivalent to setting $\tilde{x}_{0:t} = (x_{0:t-1}, \tilde{x}_t)$, i.e., the current path is not modified up to time $t-1$, when extending its dimension. Then, by substituting both (2.10) and the recursion formula (2.06) in (2.11), and also taking into account the fact that $\pi_{t-1}(x_{0:t-1}) = p(x_{0:t-1} \mid y_{1:t-1})$, it is possible to obtain the weight update equation (2.15).

$$q_t(\tilde{x}_{0:t} \mid x_{0:t-1}) = \delta(\tilde{x}_{0:t-1} - x_{0:t-1}) \cdot q_t(\tilde{x}_t \mid x_{0:t-1}) \tag{2.14}$$

$$
\begin{aligned}
w(\tilde{x}_{0:t}) &= \frac{\pi_t(\tilde{x}_{0:t})}{q_t(\tilde{x}_{0:t})} = \frac{\pi_t(\tilde{x}_{0:t})}{\pi_{t-1}(x_{0:t-1}) \cdot q_t(\tilde{x}_t \mid x_{0:t-1})} \\
&\propto \frac{\pi_{t-1}(x_{0:t-1}) \cdot p(y_t \mid \tilde{x}_t)\, p(\tilde{x}_t \mid x_{0:t-1})}{\pi_{t-1}(x_{0:t-1}) \cdot q_t(\tilde{x}_t \mid x_{0:t-1})} \\
&\propto \frac{p(y_t \mid \tilde{x}_t)\, p(\tilde{x}_t \mid x_{0:t-1})}{q_t(\tilde{x}_t \mid x_{0:t-1})}
\end{aligned}
\tag{2.15}
$$

Expression (2.15) is of great significance in the implementation of the sequential algorithm. It provides not only a theoretical framework to find an optimal importance distribution according to a predefined minimization criterion, but it also sets the foundation for the most basic SMC implementation, the sequential importance sampling (SIS) particle filter.

The SIS particle filter is implemented as follows. Within the nonlinear Bayesian filtering framework, set the importance distribution as the *a priori* pdf for the state, i.e., $q_t(\tilde{x}_{0:t} \mid x_{0:t-1}) = p(\tilde{x}_t \mid x_{t-1}) = f_t(\tilde{x}_t \mid x_{t-1})$. Thus, equation (2.15) is reduced to

$$
w(\tilde{x}_{0:t}) \propto p(y_t \mid \tilde{x}_t) = h_t(y_t \mid \tilde{x}_t) . \tag{2.16}
$$

That is, the weights for the newly generated particles are directly evaluated from the likelihood of the new observation. Although this procedure is simple, it presents severe degeneracy problems when implemented, especially if the dimensionality of the problem is large, since the variance of the weights can only increase over time [1], an issue that is solved via the implementation of an importance resampling scheme, which will be discussed in Section 2.3.2.

Although the derivation of (2.16) that is presented here assumes that (2.09) holds, it is fairly easy to generalize these results when only $q_{t-1}(x_{0:t-1})$ is available. Indeed, considering that (2.14) holds and that the importance density function at time t admits $q_{t-1}(x_{0:t-1})$ as marginal distribution at time $t-1$, i.e.

$$q_t(\tilde{x}_{0:t}) = q_{t-1}(x_{0:t-1}) \cdot q_t(\tilde{x}_t \mid x_{0:t-1}), \tag{2.17}$$

then equation (2.15) can be written as:

$$\begin{aligned} w(\tilde{x}_{0:t}) &= \frac{\pi_t(\tilde{x}_{0:t})}{q_t(\tilde{x}_{0:t})} = \frac{\pi_t(\tilde{x}_{0:t})}{q_{t-1}(x_{0:t-1}) \cdot q_t(\tilde{x}_t \mid x_{0:t-1})} \\ &\propto \frac{\pi_{t-1}(x_{0:t-1})}{q_{t-1}(x_{0:t-1})} \cdot \frac{p(y_t \mid \tilde{x}_t)\, p(\tilde{x}_t \mid x_{0:t-1})}{q_t(\tilde{x}_t \mid x_{0:t-1})} \cdot \\ &\propto w(x_{0:t-1}) \cdot \frac{p(y_t \mid \tilde{x}_t)\, p(\tilde{x}_t \mid x_{0:t-1})}{q_t(\tilde{x}_t \mid x_{0:t-1})} \end{aligned} \tag{2.18}$$

Finally, in the particular case when $q_t(\tilde{x}_{0:t} \mid x_{0:t-1}) = p(\tilde{x}_t \mid x_{0:t-1}) = f_t(\tilde{x}_t \mid x_{t-1})$, (2.19) holds and the weights for the next iteration step can be computed as in (2.20).

$$w(\tilde{x}_{0:t}) \propto w(x_{0:t-1}) \cdot p(y_t \mid \tilde{x}_t) = w(x_{0:t-1}) \cdot h_t(y_t \mid \tilde{x}_t) \tag{2.19}$$

$$w(\tilde{x}_t^{(i)}) = w_{t-1}^{(i)} \cdot h_t(y_t \mid \tilde{x}_t^{(i)}) \quad \text{and} \quad w_t^{(i)} = \frac{w(\tilde{x}_t^{(i)})}{\sum_{i=1}^{N} w(\tilde{x}_t^{(i)})} \tag{2.20}$$

The choice of the importance density function is critical for the performance of the particle filter scheme and hence, it should be considered as a parameter in the filter design. It is important to note that the update equation (2.20) is not always the best option to implement a nonlinear filtering framework. In this sense, several approaches geared to improve the performance of the algorithm, which are mainly based on the minimization of the evolution of the weight variance over time, have been suggested by different authors [1], [4], [6]-[10]. The ones that are more relevant to the objectives of the present research are discussed in Section 2.4.

2.3.2. Resampling step: the SIR Particle Filter

One of the main difficulties that must be addressed in the implementation of SIS particle filters is the degeneracy problem in the particle population. The degeneracy phenomenon consists of the fact that, as the algorithm evolves in time, the weight variances increase [11] and the importance weight distribution becomes progressively more skewed, at the point where (after a few iterations) all but one particle have negligible weights [1], [4]-[5].

As a result, the approximation of the *target* distribution becomes very poor and significant computational resources are spent trying to update particles with minimum relevance in FDI or prognosis routines. Since the degeneracy in the particle population is directly related to the variance of the importance weights, several authors have proposed to measure it using an estimate $\hat{N}_{eff}$ of the effective sample size N_{eff} [4], [9], [12].

$$N_{eff} = \frac{N}{1 + var_{\pi(\cdot|y_{0:t})}(w_{0:t})} \quad , \quad \hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N} (w_t^{(i)})^2} \tag{2.21}$$

Whenever $\hat{N}_{eff} \leq N_{thres}$, where N_{thres} is a fixed threshold, a resampling algorithm [1], [4], [13] is performed to eliminate particles with small weights and concentrate the computational efforts on those having large ones. The procedure itself generates a new set of particles $\{\breve{x}_{0:t}^{(i)}\}_{i=1\cdots N}$ by sampling N times from (2.13) such that $P(\breve{x}_{0:t}^{(i)} = x_{0:t}^{(j)}) = w_t^{(j)}$. Thus, $N_t^{(i)} \in \mathbb{N}\,(i = 1, \cdots, N)$ offspring are created for each particle, with $\sum_{i=1}^{N} N_t^{(i)} = N$. After the resampling procedure is completed, the new particle population $\{\breve{x}_{0:t}^{(i)}\}_{i=1\cdots N}$ is an i.i.d. sample of the empirical distribution (2.22) and therefore the weights are reset to $\breve{w}_t^{(j)} = N^{-1}$.

$$\breve{\pi}_t^N(x_{0:t}) = \frac{1}{N} \sum_{i=1}^{N} N_t^{(i)} \delta(x_{0:t} - \tilde{x}_{0:t}^{(i)}) = \frac{1}{N} \sum_{i=1}^{N} \delta(x_{0:t} - \breve{x}_{0:t}^{(i)}) \tag{2.22}$$

The side effect of a resampling technique is that, theoretically, the predicted paths for the state vector are no longer statistically independent. Although this condition may suggest that convergence results for the SIS algorithm are no longer valid, [6] and [15] have already established a solid theoretical foundation that proves convergence even in the SIR case.

The following algorithm summarizes the procedure [4]:

Sequential Importance Sampling Resampling (SIR) Particle Filter

1. Importance Sampling

- For $i = 1, \cdots, N$, sample $\tilde{x}_t^{(i)} \sim \pi(x_t \mid \tilde{x}_{0:t-1}^{(i)}, y_{0:t})$ and set $\tilde{x}_{0:t}^{(i)} \triangleq (x_{0:t-1}^{(i)}, \tilde{x}_t^{(i)})$.
- Evaluate the importance weights

$$w(\tilde{x}_{0:t}^{(i)}) = w_{0:t-1}^{(i)} \cdot \frac{p(y_t \mid \tilde{x}_t^{(i)})\, p(\tilde{x}_t^{(i)} \mid x_{0:t-1}^{(i)})}{q_t(\tilde{x}_t^{(i)} \mid x_{0:t-1}^{(i)})} \tag{2.23}$$

$$w_{0:t}^{(i)} = \frac{w(\tilde{x}_{0:t}^{(i)})}{\sum_{i=1}^{N} w(\tilde{x}_{0:t}^{(i)})} \tag{2.24}$$

2. Resampling Algorithm

If $\hat{N}_{eff} \geq N_{thres}$

- $\breve{x}_{0:t}^{(i)} = \tilde{x}_{0:t}^{(i)}$ for $i = 1, \cdots, N$

otherwise

- For $i = 1, \cdots, N$, sample an index $j(i)$ distributed according to a discrete distribution satisfying $P(j(i) = l) = w_t^{(l)}$ for $l = 1, \cdots, N$.
- For $i = 1, \cdots, N$, $\breve{x}_{0:t}^{(i)} = \tilde{x}_{0:t}^{j(i)}$ and $\breve{w}_t^{(i)} = N^{-1}$

In addition to the algorithm presented here, there are other versions intended to perform particle resampling, including multinomial sampling [16], residual resampling [17]-[18], and minimum variance resampling [19]. All of these algorithms ensure that

$E[N_t^{(i)}] = N \cdot w_{0:t}^{(i)}$, although they differ in the $var(N_t^{(i)})$. Residual sampling is of particular interest since it is computationally cheaper than the classical SIR technique and it offers a smaller variance for $N_t^{(i)}$, the resulting number of offspring obtained from particle (*i*).

Residual resampling involves mainly three steps [17], [20], the first being to compute $\tilde{N}_t^{(i)} = \lfloor N \cdot w_{0:t}^{(i)} \rfloor$ samples for each particle *i*. Secondly, an SIR procedure is performed to select the remaining $\bar{N} = N - \sum_{i=1}^{N} \tilde{N}_t^{(i)}$ particles, assigning them the new weights $w_{0:t}^{\prime(i)} = \bar{N}^{-1} \left(w_{0:t}^{(i)} N - \tilde{N}_t^{(i)} \right)$. The newly generated samples are then added to the current $\tilde{N}_t^{(i)}$ particles, which completes the third and last step. For this procedure, $var(N_t^{(i)}) = \bar{N}\, w_{0:t}^{\prime(i)} \left(1 - w_{0:t}^{\prime(i)}\right)$.

Other approaches introduce the use of Markov Chain Monte Carlo (MCMC) methods [21], the most famous being both the Metropolis-Hastings (MH) algorithm and the Gibbs sampler (which is a particular case of MH), as an additional step after the resampling procedure [12].

Although these methods perform very well in off-line applications, they usually are not suitable for on-line or recursive estimation because of their considerable computational requirements [3], [22]. For that particular reason, these techniques will not be discussed here.

Regardless of the method used to solve the degeneracy problem explained above, SIR Particle Filtering is a very useful and easily implementable approach that may be applied to face the problem of nonlinear filtering with a reduced dimensionality of the state-space vector. Nevertheless it is important to consider the contributions from many authors to improve the efficiency and efficacy of the algorithm, especially by defining a proper importance density function. The most relevant results are described next.

2.4. Improved Sequential Monte Carlo Methods

2.4.1. Optimal Importance Density Function

As mentioned in Sections 2.3.1 and 2.3.2, the efficiency of the importance sampling procedure improves considerably when the variance of the importance weights – conditional upon the simulated trajectory $x_{0:t-1}^{(i)}$ and the observations $y_{1:t}$ – is minimized, hence reducing the effect of the degeneracy problem. This conditional variance can be easily computed from (2.23), obtaining (2.25).

$$\operatorname{var}_{q\left(\tilde{x}_t \mid x_{0:t-1}^{(i)}, y_{1:t}\right)}\left[w\left(\tilde{x}_{0:t}^{(i)}\right)\right] = \left(w_{t-1}^{(i)}\right)^2 \cdot \left[\int \frac{\left(p\left(y_t \mid \tilde{x}_t\right) \cdot p\left(\tilde{x}_t \mid x_{t-1}^{(i)}\right)\right)^2}{q\left(\tilde{x}_t \mid x_{0:t-1}^{(i)}, y_{1:t}\right)} d\tilde{x}_t - p^2\left(y_t \mid x_{t-1}^{(i)}\right)\right] \quad (2.25)$$

It is straightforward to check that the importance weight variance (2.25) is zero when $q\left(\tilde{x}_t \mid x_{0:t-1}^{(i)}, y_{1:t}\right) = p\left(\tilde{x}_t \mid x_{0:t-1}^{(i)}, y_{1:t}\right)$, the optimal importance density function. This function was first introduced by [23] and utilized more recently in [7] and [9]. If used, it leads to the importance weight update equation:

$$w(\tilde{x}_t^{(i)}) = w_{t-1}^{(i)} \cdot p(y_t \mid x_{t-1}^{(i)}) \quad \text{and} \quad w_t^{(i)} = \frac{w(\tilde{x}_t^{(i)})}{\sum_{i=1}^{N} w(\tilde{x}_t^{(i)})}. \tag{2.26}$$

There are two major implementation problems related to the use of the optimal importance density function. First, it requires drawing samples from the distribution $p(\tilde{x}_t \mid x_{t-1}^{(i)}, y_{1:t})$. But more importantly, it involves the computation of the probability integral $p(y_t \mid x_{t-1}^{(i)}) = \int p(y_t \mid \tilde{x}_t) p(\tilde{x}_t \mid x_{t-1}^{(i)}) d\tilde{x}_t$, which does not have an analytic closed-form expression in most cases. It is important to note, however, that there is an important class of problems where these difficulties may be overcome: the Gaussian state-space model with nonlinear transition equation:

$$\begin{aligned} x_t &= f(x_{t-1}) + \omega_t, \quad \omega_t \sim \mathcal{N}(\mathbf{0}_{n_\omega \times 1}, \Sigma_\omega) \\ y_t &= C \cdot x_t + v_t, \qquad v_t \sim \mathcal{N}(\mathbf{0}_{n_v \times 1}, \Sigma_v) \end{aligned}, \tag{2.27}$$

where $f : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$ is a real valued function, $C \in \mathbb{R}^{n_y \times n_x}$ is a constant matrix, and the additive noises ω_t and v_t are mutually uncorrelated i.i.d. Gaussian sequences. In fact, for this particular case, the importance weight update equation is as follows:

$$\begin{aligned} w(\tilde{x}_t^{(i)}) &= w_{t-1}^{(i)} \cdot \exp\left(-\frac{1}{2}\left[\left(y_t - C \cdot f(x_{t-1}^{(i)}) \right)^T \cdot \left(\Sigma_\omega + C \Sigma_v C^T \right)^{-1} \cdot \left(y_t - C \cdot f(x_{t-1}^{(i)}) \right) \right] \right) \\ \text{and} \quad w_t^{(i)} &= \frac{w(\tilde{x}_t^{(i)})}{\sum_{i=1}^{N} w(\tilde{x}_t^{(i)})} \end{aligned}. \tag{2.28}$$

Suboptimal procedures based on linearization techniques (similar to the ones used in the Extended Kalman filter) have been proposed in [4] for cases when the observation equation is of the form $y_t = h(x_t) + v_t$, $v_t \sim \mathcal{N}(\mathbf{0}_{n_v \times 1}, \Sigma_v)$. The linearization procedure then ensures asymptotic convergence in the SMC algorithm when approximating the observation equation by

$$y_t \simeq h(f(x_{t-1})) + \left.\frac{\partial h(x_t)}{\partial x_t}\right|_{x_t = f(x_{t-1})} \cdot \left(x_t - f(x_{t-1})\right) + v_t \,. \tag{2.29}$$

Other authors have proposed techniques that do not necessarily rely on linearization procedures. Among the most significant ones are the following:

- **Auxiliary particle filter (ASIR)** [13]**:** This approach modifies the particle population at time $t-1$ according to the current measurement y_t, before applying the transition kernel. By doing so, it ensures that the newly created particles at time $t-1$, and hence at time *t* as well, are most likely to be close to the true state. It often provides better state estimates and is less sensitive to outliers if the process noise is small or whenever the likelihood is situated in one of the prior tails. On the other hand, the ASIR particle filter will degrade performance if the process noise is large.

- **Rao-Blackwellised particle filter (RBPF)** [6], [8]**:** This approach reduces the size of the state-space by marginalizing out some variables analytically; i.e., by considering that the posterior state pdf can be written as

$p(x_{0:t}, z_{0:t} \mid y_{1:t}) = p(x_{0:t} \mid z_{0:t}, y_{1:t})\, p(z_{0:t} \mid y_{1:t})$. The RBPF is useful when trying to estimate the *posterior* distribution of a continuous-valued state x_t as a stochastic finite mixture of Gaussians.

- **Unscented particle filter (UPF)** [20]**:** This approach considers the implementation of an unscented Kalman filter (UKF) to approximate the optimal importance density function as a Gaussian pdf, thus minimizing errors in the covariance matrix estimate and higher order moments of the *posterior*; e.g., kurtosis. In general, UPF offers better performance when the likelihood is peaked or when measurement data contains outliers.

2.4.2. Regularized Particle Filter

Although resampling schemes help to reduce the degeneracy problem, they cannot completely avoid the loss of diversity among particles since the samples are still drawn from a discrete distribution instead of a continuous one. To solve this issue, a modified approach – the regularized particle filter (RPF) – was first proposed in [24] and then reviewed in [1].

From a general perspective, the RPF is almost identical to the SIR particle filter, except for the resampling step. Whereas in the SIR particle filter the resampling step is based on the discrete approximation (2.13) for the *posterior* density $p(\tilde{x}_{0:t} \mid y_{1:t})$, the RPF considers a continuous approximation given by

$$p(\tilde{x}_{0:t} \mid y_{1:t}) \approx p^N(\tilde{x}_{0:t} \mid y_{1:t}) = \sum_{i=1}^{N} w_{0:t}^{(i)} K_h(x_{0:t} - \tilde{x}_{0:t}^{(i)})$$
$$K_h(x) = \frac{1}{h^{n_x}} K\left(\frac{x}{h}\right) \quad , \tag{2.30}$$

where $K_h(\cdot)$ is a rescaled kernel density $K(\cdot)$, $h > 0$ is the kernel bandwidth (scalar), and n_x is the dimension of the state vector. The kernel function is a symmetric pdf such that $\int xK(x)\,dx = 0$ and $\int \|x\|^2 K(x)\,dx < \infty$. Both the kernel and the bandwidth are selected to minimize (2.31).

$$MISE(p^N) = E\left[\int \left(p^N(\tilde{x}_{0:t} \mid y_{1:t}) - p(\tilde{x}_{0:t} \mid y_{1:t})\right)^2 d\tilde{x}_{0:t}\right] \tag{2.31}$$

In the special case of classically equally weighted samples, $w_{0:t}^{(i)} = N^{-1},\ i = 1, \cdots, N$, the optimal choice of the kernel is the Epanechnikov kernel [24].

$$K_{opt}(x) = \begin{cases} \dfrac{n_x + 2}{2c_{n_x}}\left(1 - \|x\|^2\right) & \text{if } \|x\| < 1 \\ 0 & \text{otherwise} \end{cases} \quad , \tag{2.32}$$

where c_{n_x} is the volume of the unit sphere in $\mathbb{R}^{n_x}$. Furthermore, if the density is Gaussian with unit covariance matrix, the optimal bandwidth is given by [24]

$$h_{opt} = A \cdot N^{-\frac{1}{n_x+4}}$$
$$A = \left(8 c_{n_x}^{-1} \cdot (n_x + 4) \cdot \left(2\sqrt{\pi} \right)^{n_x} \right)^{\frac{1}{n_x+4}} . \tag{2.33}$$

In the case of an arbitrary underlying density, two approximations are made. The first assumes that the density is Gaussian, and the second considers its covariance matrix S_t equal to the empirical covariance matrix of $\tilde{x}_{0:t}$. However, even in the general case, a suboptimal filter may be obtained by replacing the second step of the SIR particle filter implementation with the following algorithm [24].

RPF Resampling Step

If $\hat{N}_{eff} \le N_{thres}$

- Calculate S_t, the empirical covariance matrix of $\left\{ \tilde{x}_{0:t}^{(i)}, w_{0:t}^{(i)} \right\}_{i=1}^{N}$
- Compute D_t such that $D_t D_t^T = S_t$
- For $i = 1, \cdots, N$, sample an index $j(i)$ distributed according to a discrete distribution satisfying $P(j(i) = l) = w_t^{(l)}$ for $l = 1, \cdots, N$.
- For $i = 1, \cdots, N$, $\breve{x}_{0:t}^{(i)} = \tilde{x}_{0:t}^{j(i)}$ and $\breve{w}_t^{(i)} = N^{-1}$.
- For $i = 1, \cdots, N$, draw $\varepsilon^i \sim K$, the Epanechnikov kernel and $\breve{x}_{0:t}^{(i)*} = \breve{x}_{0:t}^{(i)} + h_{opt} D_t \varepsilon^i$.

Although the complexity of the RPF algorithm increases considerably when the dimensionality of the state vector is large, it is well suited for state estimation purposes

when the process noise is assumed to be independent and uncorrelated. In practice, the performance of RPF is better than the SIR particle filter when the process noise is small, although the samples are no longer guaranteed to asymptotically approximate the *posterior* pdf $p(\tilde{x}_{0:t} \mid y_{1:t})$ [24].

2.4.3. Model Parameter Estimation using Particle Filters

A framework for the estimation of fixed model parameters can be obtained as an extension of the classical particle filter approach. In fact, consider a combined set of samples $\left\{x_{0:t}^{(i)}, \theta_t^{(i)}\right\}_{i=1}^{N}$ and associated weights $\left\{w_t^{(i)}\right\}_{i=1}^{N}$ representing the posterior $p(x_{0:t}, \theta \mid y_{1:t})$ for both the state vector x and model parameter vector θ. It is important to note that the suffix t in $\theta_t^{(i)}$ indicates that the particle corresponds to the posterior pdf at time t and does not imply that θ is a time-varying parameter. From Bayes' theorem, the following proportionality relationship can be obtained for the posterior at time t:

$$\begin{aligned} p(x_{0:t}, \theta \mid y_{1:t}) &\propto p(y_{1:t} \mid x_{0:t}, \theta)\, p(x_{0:t}, \theta \mid y_{1:t-1}) \\ &\propto p(y_{1:t} \mid x_{0:t}, \theta)\, p(x_{0:t} \mid \theta, y_{1:t-1})\, p(\theta \mid y_{1:t-1}) \end{aligned}. \qquad (2.34)$$

Nevertheless, it is not clear in (2.34) how to define $p(\theta \mid y_{1:t-1})$. In this sense, two approaches are found in the literature to contend with this issue. Both of them have to deal with the problem of particle degeneracy, which is especially acute when using particle filter-based techniques for the estimation of fixed model parameters as part of an extended state vector [25].

"Artificial evolution" in [16] incorporates small random perturbations to all the parameter particles before evolving to the next time instant, as it is shown in (2.35). The update equation here considers the parameters as if they were, in fact, time-evolving. This approach has some drawbacks, the major one being that the parameters are, by assumption, fixed and hence there is an inherent loss of information in both the time and measurement updates.

$$\begin{aligned} \theta_t &= \theta_{t-1} + \zeta_{t-1} \\ \zeta_t &\sim \mathcal{N}(0, \mathbf{W}_t) \end{aligned} \tag{2.35}$$

On the other hand, [26] and [27] present kernel smoothing methods that gave a theoretical foundation for effective importance sampling techniques of $p(\theta \mid y_{1:t-1})$. Here $\bar{\theta}_{t-1}$ and $\mathbf{V}_{t-1}$ define respectively the Monte Carlo mean and variance for $p(\theta \mid y_{1:t-1})$, assuming a set of samples $\left\{\theta_{t-1}^{(i)}\right\}_{i=1}^{N}$ and weights $\left\{w_{t-1}^{(i)}\right\}_{i=1}^{N}$ that approximate $p(\theta \mid y_{1:t-1})$. The smooth kernel density is then given by

$$p(\theta \mid y_{1:t-1}) \approx \sum_{i=1}^{N} w_{t-1}^{(i)} \mathcal{N}\left(\theta \mid m_{t-1}^{(i)}, h^2 \mathbf{V}_{t-1}\right), \tag{2.36}$$

where $\mathcal{N}\left(\cdot \mid m, \mathbf{S}\right)$ is a multivariable normal density with mean *m* and covariance matrix **S,** and $h > 0$ is a smoothing parameter. On the other hand, $m_{t-1}^{(i)}$ are specified by the following rule:

$$\begin{aligned} m_{t-1}^{(i)} &= a\theta_{t-1}^{(i)} + (1-a)\bar{\theta}_{t-2} \\ a &= \sqrt{1-h^2} \end{aligned}. \tag{2.37}$$

By condensing the elements presented in [26]-[27], [25] offers a general algorithm suitable for fixed model parameter estimation with an APF implementation.

Combined State and Fixed Model Parameter Estimation

- For $i = 1, \cdots, N$, compute

$$\begin{aligned} \mu_t^{(i)} &= E[x_t \mid x_{t-1}^{(i)}, \theta_{t-1}^{(i)}] \\ m_{t-1}^{(i)} &= a\theta_{t-1}^{(j)} + (1-a)\bar{\theta}_{t-2} \end{aligned},$$

  where a is defined as in (2.37)

- Sample an auxiliary integer variable $k \in \{1, \cdots, N\}$ such that:

$$\Pr(k = i) \propto w_{t-1}^{(i)}\, p(y_t \mid \mu_t^{(i)}, m_{t-1}^{(i)})$$

- Sample a new parameter vector $\theta_t^{(k)}$ from the k^{th} component of the kernel density.

$$\theta_t^{(k)} \sim \mathcal{N}(\cdot \mid m_{t-1}^{(k)}, h^2 \mathbf{V}_{t-1})$$

- Sample $x_t^{(k)} \sim p(\cdot \mid x_{t-1}^{(i)}, \theta_t^{(i)})$

- Evaluate the importance weight

$$w_t^{(k)} \propto \frac{p(y_t \mid x_t^{(i)}, \theta_t^{(i)})}{p(y_t \mid \mu_t^{(i)}, m_{t-1}^{(i)})}$$

2.5. Particle Filtering in Real-Time Diagnosis Applications

Particle filtering, as any Bayesian technique for state estimation, has a direct application in the arena of fault detection and isolation (FDI), as well as in fault identification. Indeed, once the current state of the system is approximately known, it is natural to implement FDI procedures by comparing the process behavior with patterns regarding normal or faulty conditions.

Several authors have made use of the capabilities of SIS to complement or improve the efficiency of classical FDI approaches. Consider, for instance, the combination of particle filters and a model-based residual analysis for fault detection (FD) introduced in [28]. In that case, the PF state estimates are used to compute a residual (also referred to as the one-step prediction error), which basically indicates how far the system is from its expected (or desired) behavior. The residual signal is then employed to compute the likelihood of a faulty condition, given the assumption of additive Gaussian measurement noise.

Although this approach may be followed to improve a number of statistical tests currently available for fault detection, most of these tests rely on Gaussian additive noise assumptions to establish a closed-form expression for the evaluation of the likelihood, or for purposes of residual statistical analysis. However, many complex processes may not hold to this assumption and therefore the application of similar FD schemes may lead to inaccurate conclusions.

To solve this problem, a slightly different FDI approach is followed in [29]. Here, $M+1$ models are provided to describe the system behavior under M different fault conditions (M = 0 representing the normal mode) and a particle filtering routine is performed to obtain the state pdf estimate. The joint likelihood of the observations, conditional on each hypothesized model, is then computed as the sum of M logarithms of the Likelihood Ratio (LLR) for H_m ($m = 1, 2, \cdots, M$) versus H_o (normal operation). This paper also includes the concept of testing several plant models in parallel, which is extremely useful for the FDI approach proposed here.

This methodology, although very useful in certain application domains, does not explicitly provide statistical confidence levels or an estimate for the *type II* detection error, two of the most important customer specifications desired in a FDI routine. In fact, the FDI decision basically depends here on a fixed threshold for the sum of LLR (which will definitely vary for different applications) and no suggestions are made about how to include the already mentioned customer specifications in the design of the FDI module.

A different approach for FDI – based on the capability of particle filters to discriminate between discrete states – is followed in [30] with excellent results in the domain of propulsion systems. Hybrid dynamic models are used in [30] to represent the operation of the plant for a set of known fault modes. Discrete states in each particle represent the operational mode, while continuous-valued states describe the evolution in time of process variables. The mode of the system is computed at every time step by considering the one found most likely within the particle population.

A similar approach is utilized in [8] (Rao-Blackwellised particle filter), where the continuous-state models associated with each operational mode are considered to be linear with Gaussian additive noise. Accordingly, particle filters are only used in [8] to estimate the probabilities of each fault mode, not for continuous-valued state estimation purposes (where a simple implementation of a bank of Kalman filters suffices). The assumption of linear models, though, may be too restrictive for a general FDI framework and therefore it may not always be applicable.

Approaches introduced in [8] and [30] make use of particle filtering not only as a tool for state estimation, but also as a means of obtaining the probability of a determined fault mode in a system. This attribute is also found in other interesting results published in the literature [10], [31]-[32] and it is of paramount importance for the present research work, since it lays the foundation for including customer specifications in the design.

In this sense, two applications of particle filter algorithms for FDI purposes are of particular interest: the variable resolution particle filter (VRPF) and the risk-sensitive particle filter (RSPF). Both approaches are now described in detail.

2.5.1. Variable Resolution Particle Filters

The variable resolution particle filter [10], [32] incorporates the concept of "abstract particles" in Markov Chain processes, where each particle may represent a single state of the system or a set of similar states (also referred to as an abstract-state). This algorithm has the advantage that only a limited number of particles are needed to

represent large portions of the state-space when measurements indicate that the likelihood is low. Moreover, once the likelihood of an abstract particle increases, it is possible to specialize the associated abstract-state into a new set of individual states to represent in a better way the operational condition.

A bias-variance trade-off is performed to determine the appropriate resolution for the abstract-states, since the loss l from a particle-based approximation of the true distribution is directly related to these terms:

$$l = E\left[p(x_t \mid y_t) - \tilde{p}^N(x_t \mid y_t) \right]^2 = b\{\tilde{p}^N(x_t \mid y_t)\}^2 + \mathrm{var}\{\tilde{p}^N(x_t \mid y_t)\} \quad (2.38)$$

where $b\{\tilde{p}^N(x_t \mid y_t)\}$ is the bias and $\mathrm{var}\{\tilde{p}^N(x_t \mid y_t)\}$ is the variance of the pdf estimates. Now, consider a set of "physical" states $\left\{X^d\right\}_{d=1}^{d_k}$ in the Markov Chain with an associated stationary distribution $P(X^d)$, where $\sum_{d=1}^{d_k} P(X^d) = 1$, and a set of abstract-states $\left\{S^k\right\}_{k=1}^{M}$, which may include both physical and other children abstract-states [32]. As the resolution in the Markov Chain model decreases (more physical states are included in the abstract-states), the variance of the estimate also decreases, whereas its bias is increased. In this sense, the VRPF decides to move to a coarser resolution (abstract-state S^j) if the current resolution of the state-space is S^i, and the loss associated with resolution S^j is less than the loss of all its children [32].

$$b\{S^j\}^2 + \mathrm{var}\{S^j\} < \sum_{S^i \in \{children(S^j)\}} [b\{S^i\}^2 + \mathrm{var}\{S^i\}] \quad (2.39)$$

Interesting applications of the VRPF have been found in the arena of fault diagnosis in rovers and robots. In particular, a state-space model that includes both discrete and continuous states has been used to identify faults in six-wheel robots [10]. This work considered a VRPF to manage the posterior pdf for the discrete states, while a one-step look ahead UKF was implemented to approximate the optimal importance and deal with the estimates for the continuous states.

This methodology, also referred to as variable resolution unscented filter (VUF), has given excellent results in terms of reducing the number of particles needed to represent a system with numerous fault modes, although it requires the use of additional memory space and computational resources to store and process all sigma points [20] and covariance matrices associated with each of the particles.

2.5.2. Risk Sensitive Particle Filters

The risk-sensitive particle filter (RSPF) [31] incorporates a cost model in the importance distribution to generate more particles in high-risk regions of the state-space [10]. Mathematically, the importance distribution is set as

$$q\left(\tilde{d}_t, \tilde{x}_t \mid \tilde{d}_{0:t-1}^{(i)}, x_{0:t-1}^{(i)}, y_{1:t}\right) = \gamma_t \cdot r(d_t) \cdot p\left(d_t, \tilde{x}_t \mid y_{1:t}\right), \tag{2.40}$$

where d_t is a set of discrete-valued states representing fault modes, x_t is a set of continuous-valued states that describe the evolution of the system given those operating conditions, $r(d_t)$ is a positive risk function that is dependent on the fault mode, and γ_t is

a normalizing constant. This methodology has proven to be very helpful in improving the tracking of states that are critical to the performance of a six-wheel robot [10]. An important drawback of this approach, though, is that it needs the inclusion of exogenous models to evaluate and estimate the risk associated with every fault mode, a task that may prove to be difficult to implement.

2.6. Particle Filtering in Real-Time Prognosis Applications

Prognosis may be understood as the result of the procedure where long-term (multi-step) predictions – describing the evolution in time of a fault indicator – are generated with the purpose of estimating the remaining useful life (RUL) of a failing component/subsystem.

Failure prognosis plays an important role in achieving a reliable and cost effective operation. Certainly, this is of great interest in many industrial processes, including mechanical systems (e.g., automotive and aircrafts), power systems, continuous-time processes (e.g., pulp, paper and hot steel mills) and other discrete-time processes such as semiconductor manufacture and food products.

Several approaches related to prognosis may be found in literature [33]-[35]. Few of them, however, offer appropriate tools for real-time estimation of the RUL as a continuous function of time.

Consider, for instance, the most popular prognosis approach currently used in reliability studies: parameter estimation in Weibull-based risk probability density distributions [34]. Generally speaking, two major types of reliability degradation modeling can be distinguished, both of them particularly well suited for the task due to the ability of Weibull functions to adjust their shape and include time-dependant fault mechanisms.

On the one hand, the graphical reliability degradation modeling approach (based on statistical models and broadly used in practice) selects risk function parameters so that degradation data is satisfactorily represented at each observation time. Parameter estimation is performed through a two step off-line statistical procedure that greatly depends on the amount of degradation data and is subject to accumulation of computational errors [34]. Also, if the faulty system is behaving differently from what was previously observed, the method is unable to correct the predictions, which leads to inadequate prognosis results and thus inexact conclusions.

On the other hand, the degradation path curve approach intends to approximate a degradation trajectory versus time, using known physics-based models or a statistically designed path curve. Although this methodology offers better results that the graphical approach, it still lacks learning or adaptation mechanisms and must be performed off-line since it is extremely computationally intensive. Recent improvements focus on the use of maximum likelihood estimation (MLE) techniques and truncated Weibull pdf's, obtaining good results in the case of fatigue crack growth data [34], although this only

applies when degradation data follow a two-parameter Weibull distribution and this technique does not solve the problem of real-time applications.

In this sense, the most comprehensive effort in establishing an on-line prognosis framework can be found in applications associated with the use of filtering techniques for the study of fatigue crack dynamics [35]. The filtering concept enhances the deterministic crack growth modeling standpoint based on the application of Paris's Law [36]-[37], and keeps a close relationship to the physics of the problem. Efforts have been made to employ Markov processes and Extended Kalman Filters (EKF) to estimate the first two moments of a Gaussian state pdf of the system, also assuming independence between measurement noise and uncertainties in material properties. The obtained Gaussian pdf is afterwards projected in time and used to test M disjoint statistical hypothesis, which divide the feasible range for crack length values, thus obtaining M different probability distributions describing the time evolution of the likelihood for each hypothesis [35].

Although the previous approach sets the foundations for other filtering-based real-time prognosis methodologies, there are still some unsolved issues. Firstly, the assumption of a Gaussian (or log-normal) pdf is not always held in nonlinear processes and therefore the projection in time of the filtered state pdf may lead to problems both in terms of accuracy and precision. Secondly, the fragmentation of the crack length domain in M disjoint regions leading to a set of M hypothesis is not always the best prognosis procedure, since different crack length may affect the system's performance in dissimilar ways. For example, the probability of failure is not uniform for all crack lengths; there are in fact some regions where a failure condition is more frequently detected. Hence, it

would be desirable to somehow collect all the information contained in the *M* hypothesis into a single prognosis outcome for the operator, also considering the probability of failure for each partition of the crack length domain. Thirdly, the methodology presented in [35] has only been tested for a particular application and it does not offer an alternative way to be implemented in processes with unknown time-varying model parameters. Lastly, no indication is made in [35] about how to propagate a non-Gaussian state pdf in time through nonlinear transition models.

Regarding particle filters, most authors have visualized this technique (and other nonlinear filtering approaches) as a tool for detection. In the case of prognosis, though, there are no clear indications about how to project the particle population in time, while keeping the assumptions about model nonlinearities and non-Gaussian noise structures. In specific applications, such as chaos prediction, it has been suggested to assume absence of both process and measurement noise for prediction purposes [33], thus obtaining a long-term prediction with minimum variance. Each particle is then used as an initial condition for deterministic models in order to be used for decision theory, risk calculations and other statistical approaches. The implications of these assumptions, though, could be significant in real processes and therefore they must be evaluated with care.

3. PARTICLE-FILTERING-BASED FRAMEWORK FOR DIAGNOSIS IN NONLINEAR SYSTEMS

The present chapter introduces the particle-filtering-based framework for on-line fault diagnosis in nonlinear systems and shows the results obtained in two case studies where real fault data was available. The structure of the chapter is as follows. Section 3.1 provides a general description of the diagnosis framework. Section 3.2 presents a case study where the proposed methodology has been applied to detect a crack in one of the blades of a turbine high-power-compressor (HPC) disk. Section 3.3 considers the implementation of a fault diagnosis module that detects a change in the growth rate of an axial crack in an UH-60 planetary gear carrier plate and discusses the obtained results. Lastly, for completion purposes, Section 3.4 presents a description of how the particle-filtering-based methodology could be used to implement an anomaly detector. Discussion in Section 3.4 is based on the results presented in Section 3.3, which basically show how the proposed particle-filtering-based framework can be applied to distinguish between two types of operating modes: *normal* and *faulty*.

3.1. General Description of the Diagnosis Framework

A fault diagnosis procedure involves the tasks of fault detection and isolation (FDI), and fault identification (assessment of the severity of the fault). In general, this procedure may be interpreted as the fusion and utilization of the information present in a feature vector (measurements), with the objective of determining the operating condition (state) of a system and the causes for deviations from particularly desired behavioral

patterns. Several ways to categorize FDI techniques can be found in literature. Particularly in this thesis, FDI techniques are classified according to the way that data is used to describe the behavior of the system: *data-driven* or *model-based* approaches [37].

Data-driven FDI techniques usually rely on signal processing and knowledge-based methodologies to extract the information hidden in the feature vector (also referred to as measurements). In this case, the classification procedure may be performed on the basis of variables that have little (or sometimes completely lack of) physical meaning. On the other hand, *model-based* techniques, as the name implies, use a description of a system (models based on first principles or physical laws) to determine the current operating condition.

A compromise between both classes of FDI techniques is often needed when dealing with complex nonlinear systems, given the difficulty of collecting useful faulty data (a critical aspect in any *data-driven* FDI approach) and the expertise needed to build a reliable model of the monitored system (a key issue in a *model-based* FDI approach).

From a nonlinear Bayesian state estimation standpoint, this compromise between *data-driven* and *model-based* techniques may be accomplished by the use of a particle-filter-based module built upon the nonlinear dynamic state model (3.01) [10]:

$$\begin{cases} x_d(t+1) = f_b\left(x_d(t) + n(t)\right) \\ x_c(t+1) = f_t(x_d(t), x_c(t), \omega(t)) \\ \text{Features}(t) = h_t(x_d(t), x_c(t), v(t)) \end{cases}, \qquad (3.01)$$

where f_b, f_t and h_t are nonlinear mappings, $x_d(t)$ is a collection of Boolean states associated with the presence of a particular operating condition in the system (normal operation, fault type #1, #2, etc.), $x_c(t)$ is a set of continuous-valued states that describe the evolution of the system given those operating conditions, $\omega(t)$ and $v(t)$ are non-Gaussian distributions that characterize the process and feature noise signals respectively. Since the noise signal $n(t)$ is a measure of uncertainty associated with Boolean states, it is advantageous to define its probability density through a random variable with bounded domain. For simplicity, $n(t)$ may be assumed to be i.i.d. uniform white noise.

A particle-filter-based approach using model (3.01) allows statistical characterization of both Boolean and continuous-valued states, as new feature data (measurements) are received. As a result, at any given instant of time, this framework provides an estimate of the probability masses associated with each fault mode, as well as a pdf estimate for meaningful physical variables in the system. Once this information is available within the FDI module, it is conveniently processed to generate proper fault alarms and to report on the statistical confidence of the detection routine.

Two specific approaches are hereby proposed to generate fault alarms from the pdf estimates that are computed on-line:

- The first approach defines the outputs of the FDI module as the expectations of the Boolean states in model (3.01), providing a recursively updated estimate of the probability for each fault condition considered in the analysis. These expectations may activate alarm indicators or prognostic modules if they exceed

appropriate thresholds for the probability of detection (typically 90% or 95%). This approach is particularly useful when the normal operation of the system is defined through a dynamic state-space model.

- The second approach defines the output of the FDI module as the statistical confidence needed to declare the fault via hypothesis testing. This test is performed between the pdf estimate of one of the continuous-valued states in model (3.01) and another pdf defining the desired condition (baseline). This approach allows including variables with physical meaning into the decision-making procedure. Additionally, it is very useful when diagnosing deviations from a particular setpoint, since historical data can be used to build the baseline pdf. Other indices, such as Fisher's discriminant ratio, may also be used to distinguish between the aforementioned probability distributions as long as there exists a criterion to define a threshold to declare the fault condition.

These approaches may be either used separately or in a combined manner within the proposed diagnosis framework to provide a satisfactory solution for the FDI problem. Each of these methodologies is described, illustrated, and discussed in the application examples provided in Sections 3.2 and 3.3.

It is important to mention that the proposed fault diagnosis framework also allows for the use of the pdf estimates for the system continuous-valued states (computed at the moment of fault detection) as initial conditions in the failure prognostic routines, giving excellent insight into the inherent uncertainty in the prediction problem. As a result, a

swift transition between the two modules (fault diagnosis and failure prognosis) may be performed, and moreover, reliable prognosis can be achieved within a few cycles of operation after the fault is declared. This characteristic is, in fact, one of the main advantages offered by this particle filter-based framework.

3.2. Detection of Cracks in Blades of a Turbine HPC Disk

Consider the problem of detecting and identifying cracks in blades of a turbine high-power-compressor (HPC) disk (see Figure 3.1) in an on-line fashion. Although a *data-driven* FDI approach alone may solve the detection problem, this kind of techniques may prove to be insufficient to address the identification issue (i.e. to estimate the actual crack length in the blade), especially if afterwards it is intended to prognosticate the evolution in time of the fault condition. On the other hand, structural analysis of cracked blades may successfully address the failure prognosis problem once the actual crack length is known (using FRANC-3D or ANSYS software packages), but it is impractical to implement such an approach in an on-line manner.

Figure 3.1. Picture of turbine high-power-compressor (HPC) disk

The proposed on-line particle-filter-based FDI framework offers a means to combine both types of approaches. The built-in nonlinear dynamic model allows using concepts from structural analysis to describe the expected evolution of the crack in time, for a given load profile, and the measurement equation in that model allows to relate the current size of the crack to on-line data.

For the purpose of generating measurement data associated with the crack length, light probes on both the leading and the trailing edge of the blades have been installed to provide data associated with the time-of-arrival (TOA) of the light signal for each blade. In the absence of a crack, it is expected that the passing times for each blade will be only defined by the position of the probes, the radius of the rotor, and its rotational speed. When a crack appears in a particular blade, it generates vibrations that affect the blade passing times.

Data pre-processing techniques are needed to translate this concept into a feature that can be used for detection purposes, i.e. the tangential blade position (TBP). Figure 3.2 shows a schematic that illustrates the pre-processing steps. Firstly, a least squares algorithm estimates two parameters, namely A and B, which relate the inter-blade spacing (IBS) with the square of the existing normalized rpm value, as shown in (3.02). Secondly, the TBP feature is computed, given the fact that the rotational speed is maintained in the nominal value (i.e. the normalized speed is $rpm = 1$) using (3.03), where b represents the number of blades in the HPC disk.

$$IBS = A + B \cdot rpm^2 \tag{3.02}$$

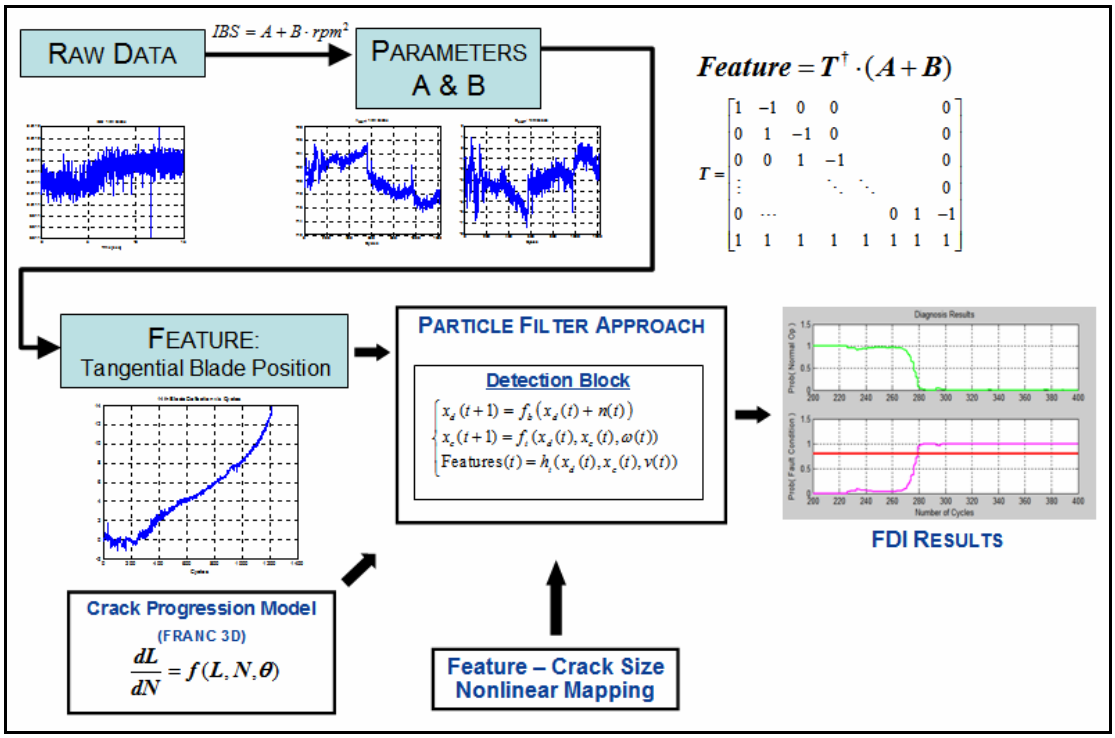

Figure 3.2. Implementation of a particle filter-based FDI module to detect and isolate cracks in blades of a turbine HPC disc

Given that the IBS between two consecutive blades is a measure of their spatial separation, then it must be equal to the difference between their correspondent TBPs. Moreover, the vector sum of all TBPs should be null. Equation (3.03) summarizes these concepts in a close-for expression [36].

$$\begin{bmatrix} TBP \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 & & & & 0 \\ 0 & 1 & -1 & 0 & & & & 0 \\ 0 & 0 & 1 & -1 & & & & \vdots \\ \vdots & & & \ddots & \ddots & & & 0 \\ 0 & \dots & & & & 0 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{\dagger}_{(b+1)\times b} \cdot (A + B) \qquad (3.03)$$

In addition to the vibration-based feature aforementioned, and as part of DARPA Prognosis program contract no. HR0011-04-C-001, an exhaustive FRANC-3D structural analysis was conducted by Dr. Tulpule Sharayu at Pratt & Whitney. This structural analysis focused on the stresses that appear on turbine blades undergoing a crack and resulted in a equation that relate the number of cycles in operation with estimate of the crack length. The author converted this equation into a time-varying state space model and added a model parameter that can be updated, as feature data is received on-line, to account for modeling errors. Equation (3.04) represents the resulting model, which is suitable for describing the growth of a crack on any blade under nominal load conditions [36], [38]:

$$\frac{dL}{dn} = \frac{1}{6\alpha \cdot L^5(n) + p(L(n))}, \tag{3.04}$$

where L is the length of the crack (in inches), n is the number of stress cycles applied to the material, α is a model parameter to be estimated and $p(L(n))$ is a known fourth order polynomial determined with the help of the FRANC-3D structural model.

Assuming the existence of a 1-to-1 nonlinear mapping $h(\cdot)$ between the feature information and the actual size of the crack in the blade, it is possible to implement a nonlinear model suitable for a particle filter-based FDI framework. The model is shown in (3.05), with β a known model parameter and $\omega(t)$ and $v(t)$ have been selected as zero mean Gaussian noise profiles for simplicity.

$$
\begin{aligned}
&\begin{bmatrix} x_{d,1}(t+1) \\ x_{d,2}(t+1) \end{bmatrix} = f_b\left(\begin{bmatrix} x_{d,1}(t) \\ x_{d,2}(t) \end{bmatrix} + n(t) \right) \\
&x_c(t+1) = \left[\left(1+\beta\right) x_c(t) \right] \cdot x_{d,2}(t) + \omega(t) \\
&\text{Feature}(t) = x_c(t) + v(t) \\
&f_b(x) = \begin{cases} \begin{bmatrix} 1 & 0 \end{bmatrix}^T \text{, if } \left\| x - \begin{bmatrix} 1 & 0 \end{bmatrix}^T \right\| \le \left\| x - \begin{bmatrix} 0 & 1 \end{bmatrix}^T \right\| \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^T \text{, otherwise} \end{cases} \\
&\begin{bmatrix} x_{d,1}(0) \\ x_{d,2}(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\end{aligned}
\qquad (3.05)
$$

In equation (3.05), $x_{d,1}$ and $x_{d,2}$ are Boolean states that indicate *normal* and *faulty* conditions, respectively. The continuous-valued state x_c represents the crack length, β is a time-varying model parameter, and $\omega(t)$ and $v(t)$ have been selected as zero mean Gaussian noise profiles for simplicity.

The FDI module itself discriminates between two Boolean states: absence of crack in any blade (*normal condition*) or presence of crack (*fault condition*) in at least one of them. The output of the FDI module has been defined in this case as the current expectation of the Boolean state associated with the fault mode, i.e. $E\left\{x_{d,2}(t)\right\}$. As one value of the tangential blade position feature is calculated per blade and per cycle, the module is also able to pinpoint the blade that has been affected by the crack condition (fault isolation). Results of the detection module for the case of one particular blade are shown in Figure 3.3.

Although it is possible to observe some changes in the probability of failure condition around the 230th cycle of operation, only after the 280th cycle there is evidence that is strong enough to ensure the existence of a crack ($E\{x_{d,2}(t)\} \geq 0.80$). After that particular time instant, the FDI task has been completed and the pdf estimate for the state $x_c(t)$ – together with the mapping $h(\cdot)$ – may be used for prognosis purposes.

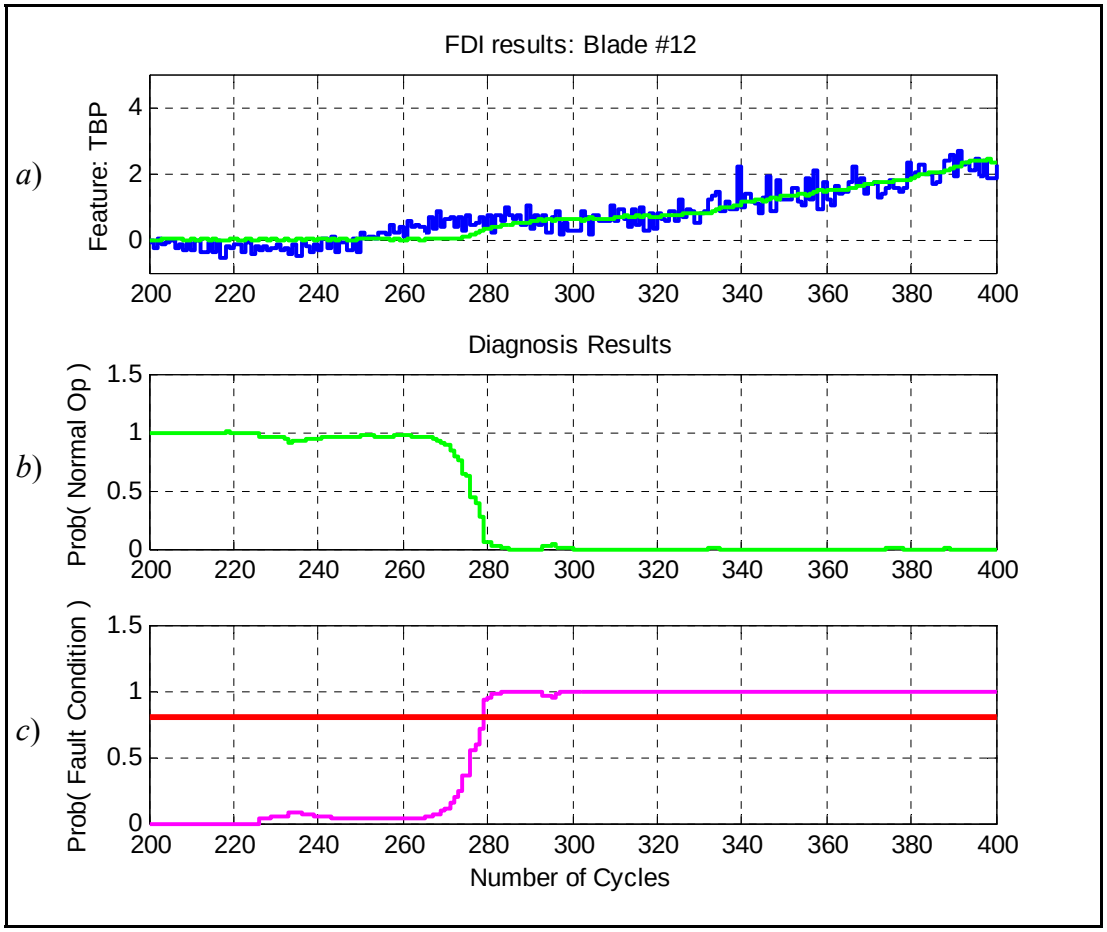

Figure 3.3. Detail of detection results for a crack in a turbine engine blade. a) Evolution in time of the TBP feature (blue) and filtered output from the FDI module for the continuous-valued state (green). b) $E\{x_{d,1}(t)\}$. c) $E\{x_{d,2}(t)\}$ and threshold of 80% for the probability of a fault (red horizontal line)

Results in this case show the efficiency of the algorithm in pinpointing the abnormal situation, combining the knowledge of structural models and real-time measurements to generate simple and reliable fault indicators.

3.3. Detection of Cracks in a UH-60 Planetary Gear Carrier Plate

The study of the growth of an axial crack in a UH-60 gear plate and the details that should be considered in the development of a structural model, needed in any on-line model-based approach for FDI and failure prognosis, are analyzed in [37].

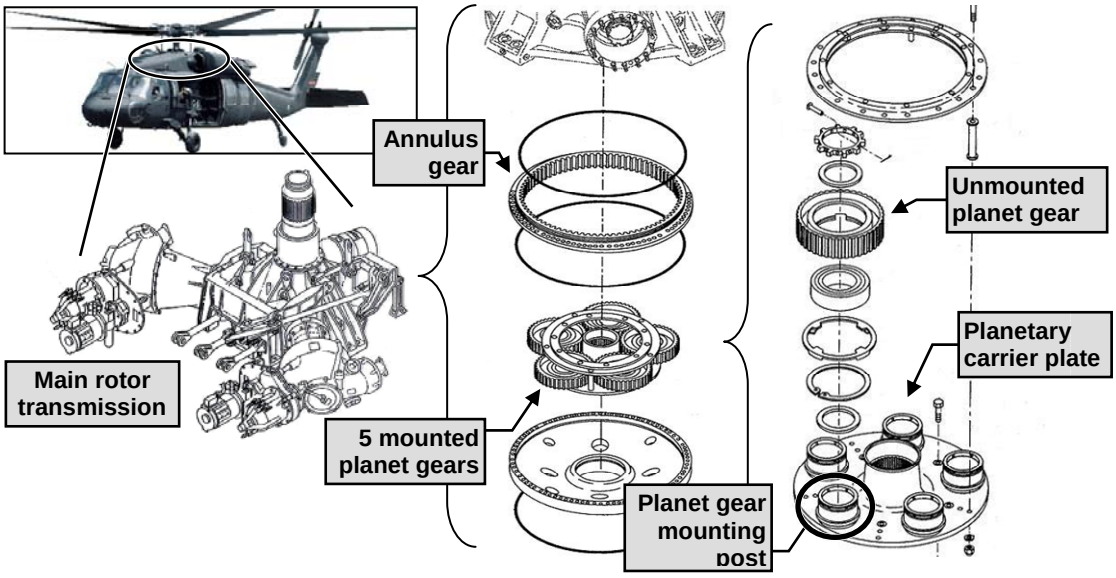

Figure 3.4. Mechanical components of the UH-60 helicopter transmission. Picture extracted from [37]

Consider the case of a seeded fault test on a carrier plate, a critical component of the *planetary gear* transmission system that transmits mechanical power from the engines to the main rotor blades of the helicopter, as shown in Figure 3.4. In this test, a cyclic

load profile is applied to the carrier plate to analyze how it affects the growth of an existing axial crack. Given a set of vibration data, it is possible to compute an approximate (and noisy) estimate of the crack length via the use of features that compute the ratio between the fundamental harmonic and the sidebands of the vibration signal spectrum. One particular example of these features is the *sideband ratio* (SBR) [37]. The relationship between vibration-based features and the actual length of the axial crack in the carrier plate becomes the foundation for the implementation of the on-line FDI module, and later on for the failure prognosis module.

Given that the existence of a fault condition is known in a seeded fault test, the main objective in this case study is to determine when this crack increases in length along its axis. Customer requirements include early detection of changes in the growth rate at a desired statistical confidence level for the alarm signal (typically 95%). Thus, two main operating conditions are distinguished: the *normal* condition reflects the fact that the crack is growing very slowly or not growing at all, meanwhile the *faulty* condition indicates an abrupt change in the growth rate.

In this case, a particle-filtering-based FDI module may be implemented using the nonlinear model (3.06) to describe the expected rate of crack growth, where $x_{d,1}$ and $x_{d,2}$ are Boolean states that indicate *normal* and *faulty* conditions, respectively, x_c is the continuous-valued state that represents the crack length, β is a time-varying model parameter dependant on the loading profile which is being applied to the gearbox, and $\omega(t)$ and $v(t)$ have been selected as zero mean Gaussian noise profiles for simplicity. The

initial crack length in the real fault data set used in this analysis (obtained from the seeded fault test) is 3.4 inches, which is reflected in the initial condition used in model (3.06). Given that the objective here is to determine the time instant when the crack increases significantly from its initial length, the initial value for the Boolean states indicates a *normal* condition for a crack of 3.4 inches.

$$
\begin{aligned}
&\begin{bmatrix} x_{d,1}(t+1) \\ x_{d,2}(t+1) \end{bmatrix} = f_b\left(\begin{bmatrix} x_{d,1}(t) \\ x_{d,2}(t) \end{bmatrix} + n(t) \right) \\
&x_c(t+1) = x_c(t) + \beta \cdot x_c(t) \cdot x_{d,2}(t) + \omega(t) \\
&y(t) = x_c(t) + v(t) \\
&\\
&f_b(x) = \begin{cases} \begin{bmatrix} 1 & 0 \end{bmatrix}^T \text{, if } \left\| x - \begin{bmatrix} 1 & 0 \end{bmatrix}^T \right\| \le \left\| x - \begin{bmatrix} 0 & 1 \end{bmatrix}^T \right\| \\ \begin{bmatrix} 0 & 1 \end{bmatrix}^T \text{, otherwise} \end{cases} \\
&\begin{bmatrix} x_{d,1}(0) \\ x_{d,2}(0) \\ x_c(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 3.4 \end{bmatrix}
\end{aligned}
\qquad (3.06)
$$

Besides detecting the *faulty* condition that has been previously defined, it is also desired to obtain some measure of the statistical confidence of the alarm signal. For this reason, it has been decided to define the output of the FDI module as the statistical confidence needed to declare the fault via hypothesis testing (H0: "The crack is not growing" vs. H1: "The crack is rapidly growing"). This test is performed between the current pdf estimate for $x_c(t)$ – in model (3.06) – and another pdf defining the desired condition (baseline).

The baseline pdf, in this case, has been defined as a Normal distribution $\mathcal{N}(\mu,\sigma^2)$, where $\mu = 3.4$. The parameter σ is an estimate of the standard deviation of the signal noise, and it can be estimated from historical data. One way to generate an indicator of statistical confidence for the detection procedure is through the computation of the *type II detection error* [39]. Specifically, it is proposed to consider the sum of the weights of all particles i such that $x_c^{(i)}(T) \geq z_{1-\alpha,\mu,\sigma^2}$, where α is the desired test confidence and T is the detection time, which is essentially equivalent to an estimate of $(1 - \textit{type II error})$, or equivalently the probability of detection. If additional information is required, it is possible to compute the value of Fisher's discriminant ratio, given in this particular case by (3.07). Once computed, either of the above mentioned indicators may help to define an appropriate detection threshold for the problem under study,

$$F_{index}(T) = \frac{\left| \mu - \sum_{i=1}^{N} w_T^{(i)} \cdot x_c^{(i)}(T) \right|^2}{\sigma^2 + \sum_{i=1}^{N} w_T^{(i)} \cdot \left(x_c^{(i)}(T) - \sum_{j=1}^{N} w_T^{(j)} \cdot x_c^{(j)}(T) \right)^2}. \qquad (3.07)$$

Figure 3.5 shows the results obtained when the proposed FDI approach is applied to the problem of crack growth detection in the planetary gear plate, using the state-space model (3.06) and 500 particles to describe its evolution in time. The vertical line that discriminates between the two pdf's in Figure 3.5 is fixed by the desired *type I* detection error (5% probability of false positives), considering the data that is used as a baseline for detection purposes. The time unit in this case is a ground-air-ground (GAG) cycle [37].

By observing the trend of the vibration-based crack estimate over time, it is clear that no significant increment in the crack length happened before the 100th GAG cycle. The algorithm only needs 35 additional GAG cycles to detect a change in the growth rate, with a confidence level (α) of nearly 70% (*type II* error $\approx 30\%$). At the same time instant, Fisher's discriminant ratio surpasses a threshold of 3.5 units.

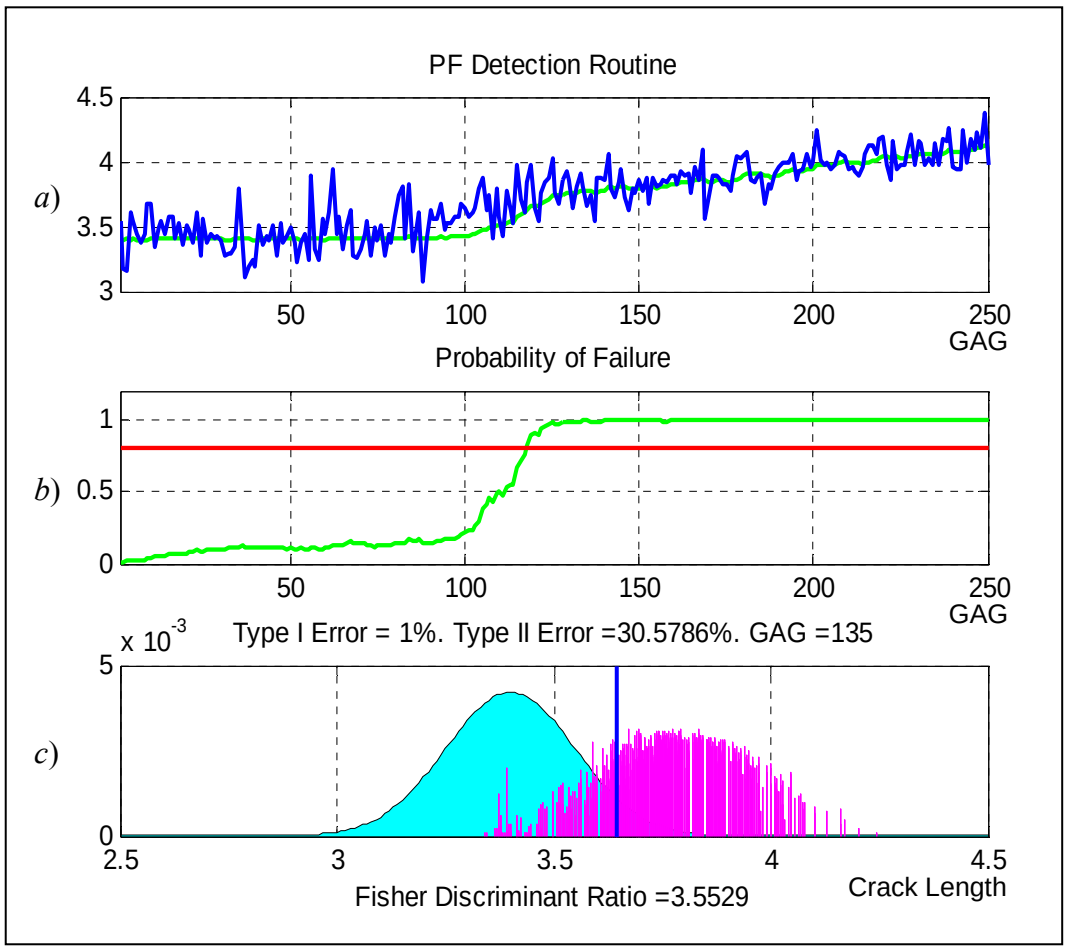

Figure 3.5. Particle filter-based FDI module. Cracked plate problem. a) Evolution in time of crack length (blue) and filtered estimate from the FDI module (green). b) $E\{x_{d,2}(t)\}$. c) Baseline pdf (cyan) and estimate for the crack length pdf (magenta), the vertical blue line indicates the threshold associated with a 5% *type I detection error* (probability of false alarms), the value of Fisher's discriminant ratio for both density functions is also indicated

Figure 3.5 shows the three fault indicators that are simultaneously computed. The first indicator, depicted as a function of time, shows the probability of a specified failure mode within the system, and it is based on the estimate of the discrete-valued state $x_{d,2}$ in model (3.06). FDI results may be obtained whenever this probability reaches the desired confidence level for a particular mode. If more information is needed, the *type II detection error* or the value of the Fisher's discriminant ratio (2nd and 3rd indicators, respectively) may be also considered.

It must be noted that, when using the 2nd indicator, there is no need to specify a detection threshold for the current values of the states in the monitored system. In this case, customer specifications are translated into acceptable margins for the *type I* and *II* errors in the detection routine. The algorithm itself will indicate when the *type II error* (probability of false negatives) has decreased to the desired level.

Finally, it is of paramount importance to compare the performance of the different proposed detection indices, considering the type of particle-filter algorithm and the number of particles that is being used in the FDI module. For this purpose, a variant of the RSPF (Section 2.5.2) has been implemented by increasing the mean value of the additive noise $n(t)$ in the dynamic model (3.06). This variant modifies the *prior* pdf in such a way that the particle-filtering algorithm facilitates the spawn of particles associated with a fault condition. In the event of a *normal* operating condition, the likelihood of the measurements will considerably diminish the importance of the particles indicating a fault (by reducing the associated weights); however, if the *faulty* condition

appears, the variant of the algorithm ensures that more particles in the population will be sensitive to the anomaly, thus reducing the detection time.

Table 3.1 summarizes results obtained when varying the number of particles (N = 30, 100, and 500) and the mean of the process noise $n(t)$. Tabled values represent the time of detection (in GAG cycles) when the threshold for the alarm indicator is settled for a specified confidence level in detection, namely $P_d = (1 - type\ II\ error)$. In addition, a last column has been added to show the performance of a detection module based on the expectation of the Boolean state associated with the *faulty* condition.

Table 3.1. Time of detection (in GAG cycles), given different thresholds for the particle-filter-based estimate of the probability of detection, where $P_d = (1 - type\ II\ error)$.

	P_d = 70%			**P_d = 90%**			**P_d = 95%**			**$E\{x_{d,2}(t)\} \geq 0.9$**		
N / **$E\{n(t)\}$**	**30**	**100**	**500**	**30**	**100**	**500**	**30**	**100**	**500**	**30**	**100**	**500**
0.00	138	145	165	181	201	200	200	210	208	123	124	122
0.05	160	162	165	177	180	200	197	207	210	127	119	118
0.10	164	154	162	194	198	196	215	206	205	109	109	115
0.50	180	157	165	187	182	186	199	212	208	4	4	4

Results listed in Table 3.1 show that the number of particles in the algorithm has no clear influence on the time of detection, as long as this number is sufficiently large to infer statistical results in the FDI module. A word of caution must be said with respect to this fact: although it would be tempting to reduce the number of particles, it is important to note that the quality of the pdf estimate for the continuous-valued states is critical to ensure a good initial condition in any prognostic routine. Thus, the proper choice for the

number of particles will be intrinsically associated with the computational resources that are available for FDI purposes.

The effect of the mean value in the process noise $n(t)$ is significant, particularly for thresholds associated with a lower probability of detection. Given that $n(t)$ affects the evolution in time of Boolean states, it is not recommended to use a mean value greater than $E\{n(t)\} = 0.5$ and, in particular for this case, the best results are obtained when $E\{n(t)\} = 0.05$. Interestingly, the effect of this parameter is reduced as the number of particles increases, which is a fact that supports the recommendation stated in the paragraph above.

Lastly, Table 3.1 shows that a fault indicator based on the expectation of a Boolean state is highly sensitive to the mean value of the process noise $n(t)$, to the extent that it completely fails to provide a reasonable answer when $E\{n(t)\} = 0.5$. It also shows that, in general, the expectation of a Boolean state is a more sensitive indicator since the time of detection is significantly smaller than that obtained from approaches based on pdf estimates for the continuous-valued states.

3.4. Detecting Unanticipated Faults: Anomaly Detector

The concept of "coverage" – in terms of the number of fault modes that a diagnosis system is capable of detecting, isolating and identifying – has recently been the subject of increased interest within the FDI community. A diagnosis system provides a solution for the problem of monitoring a finite number of fault modes that are

conveniently ranked and selected according to a Failure Modes, Effects, and Criticality Analysis (FMECA); nevertheless, it is also desired that the monitoring system has the capability to detect unanticipated faults or discrepancies with respect to expected behaviors.

From this standpoint, the objective is not to pinpoint the identity of the problem, but to simply recognize the existence of deviations from the expected (or normal) operation of the plant (or subsystem), assuming that there is a set of indicators (features) that properly characterize its most critical aspects. Although some of these features may be used for the detection of a very specific fault mode, it is desirable that most of them undergo significant perturbations in the presence of several different fault scenarios.

In this sense, an anomaly detector is a module intended to recognize abnormal conditions in the operation of a monitored system. In most real applications, the anomaly detector is required to perform this task while minimizing both the probability of false alarms and the detection time (time between the initiation of a fault and its detection), given a fixed threshold for false positives. A module with these characteristics involves a comparison between the current condition of the system and the expected operational behavior. In that sense, the availability of historical data is always assumed for purposes of defining an appropriate baseline. The present section intends, only for completion purposes, to describe how the proposed particle-filtering-based approach can be used to implement an anomaly detector module.

In particular, consider the schematic shown in Figure 3.6 depicting a simple proposed architecture for an anomaly detector. In this architecture, real-time measurements and information about the current operational mode are provided in an on-line fashion. Data are pre-processed and de-noised before computing the features that will help to efficiently monitor the behavior of the plant. Statistical analysis applied to this set of features, which compare their evolution in time with respect to the baseline data, are afterwards performed to simultaneously arrive at the probability of abnormal conditions, the probability of false alarms, and a simple on/off indicator. This indicator shows the exact time instant when the presence of a fault can be confirmed for a given confidence level (usually 95%). If time and computational resources allow further analysis, feature information can afterwards be used to complete the tasks of fault isolation, identification, and failure prognosis.

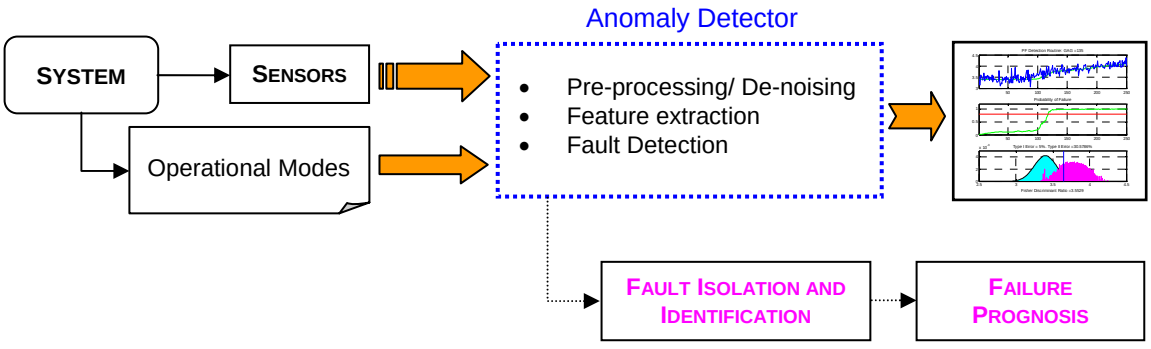

Figure 3.6. Proposed architecture for an anomaly detector. Fault isolation and identification, as well as failure prognosis, are optional task that may be performed after the anomaly is detected

The architecture proposed in Figure 3.6 fits into the particle-filter-based framework for fault detection proposed in Section 3.1, and particularly with the implementation described in Section 3.3. In fact, consider the results already shown in Figure 3.5 (c) where a baseline pdf (cyan) is compared to the current pdf estimate for the crack length (magenta). Given both the existence of a baseline pdf for the feature(s) being monitored and an estimate for the current pdf of that (those) feature(s), it is plausible to apply any of the fault alarm indices presented in Sections 3.2 and 3.3 to quantify the separation between the distribution functions, even in the case when the cause for such separation is unknown. The detection of the anomaly can be even made when the feature does not monotonically increase (or decrease) due to the fault condition, since the objective here is not to identify the fault (i.e. to assess the severity of the fault condition), but only to perceive a difference with respect to the established baseline.

The concepts stated in this section lay the foundation for a wide variety of application domains. Given that the theoretical framework presented in this chapter can be applied to detect a fault condition in a two-class problem, a novelty detection [40] procedure becomes basically a particular case of the proposed particle-filtering-based diagnosis framework.

4. PARTICLE-FILTERING-BASED FRAMEWORK FOR PROGNOSIS IN NONLINEAR SYSTEMS

4.1. General Description of the Prognosis Framework

Prognosis may be understood as the generation of long-term (multi-step) predictions describing the evolution in time of a particular signal of interest or fault indicator, with the purpose of estimating the remaining useful life (RUL) of a failing component/subsystem. Since prognosis projects the current system condition in time – using a state dynamic model in the absence of future measurements – it necessarily entails large-grain uncertainty. For this reason, any accurate and precise prognosis scheme should unavoidably consider fault indicators (and other critical state variables) as random processes in such a way that, once their probability distributions are estimated, other important attributes – such as confidence intervals – may be computed.

Real-time data (measurements or features) provided by sensors monitoring key fault parameters suggest a possible solution to the prognosis problem based on recursive Bayesian estimation techniques. With this approach, long-term predictions for the fault indicators are generated using dynamic growth models, while accurate real-time state estimates define initial conditions for those models. It is also reasonable to assume that sensor data will be available for a certain time window if the incipient failure is detected and isolated at early stages, allowing adjustments in the predictions (via updates in model parameter and state estimates) and thus improving the prognostic results both in terms of accuracy and precision [39]. At the end of the observation time window, however, the

prediction outcome should be delivered to the user (operator, maintainer), who must decide which corrective actions to take in order to avoid a catastrophic event.

Particle filtering, as it has been previously mentioned, is particularly useful when dealing with difficult nonlinear and/or non-Gaussian processes and thus is suitable as a Bayesian tool for prognosis purposes [36], [38]. Sequential importance sampling helps to reduce the number of samples required to approximate distributions with appropriate precision, being faster and more computationally efficient than classical Monte Carlo methods. Moreover, particle filtering allows information from multiple measurement sources to be fused in a logical manner.

Prognosis, though, is a problem that goes beyond the scope of filtering applications since it involves future time horizons. Hence, if PF-based algorithms are to be used, it is necessary to propose a procedure with the capability to project the current particle population in time in absence of new observations, adjusting weights if necessary. Furthermore, since the main source of uncertainty is related to the fact that both the process and measurement models (including noise definition and statistics) are subject to errors, then it is not possible to account for all the uncertainty only through the refinement of the state estimation technique. With the purpose of solving the aforementioned issues, a two-level procedure has been developed and subsequently tested. This procedure intends to reduce the uncertainty associated with long-term predictions by using the current state pdf estimate, the process noise model, and a record of corrections made to previously computed predictions.

In a first prognosis level, *p*-step ahead predictions are generated on the basis of an *a priori* estimate, adjusting probabilities that are associated with the prediction according to the noise model structure. A second prognosis level uses these predictions and the definition of critical thresholds to estimate the RUL pdf, also referred to as the time-to-failure (TTF) pdf, and simultaneously implements a correction model (*outer correction loop*) to compensate for all main error sources. A detailed description of each level is now presented.

4.2. First Prognosis Level: Generation of Long-Term Predictions

The first prognosis level is related to the generation of a *p*-step ahead long term prediction for the state pdf of a dynamic system, which can be obtained in a recursive manner using both the model update equation (2.01) and the current state estimate, as shown in (4.01).

$$\begin{aligned} \tilde{p}(x_{t+p} \mid y_{1:t}) &= \int \tilde{p}(x_t \mid y_{1:t}) \prod_{j=t+1}^{t+p} p(x_j \mid x_{j-1}) dx_{t:t+p-1} \\ &\approx \sum_{i=1}^{N} w_t^{(i)} \int \cdots \int p(x_{t+1} \mid x_t^{(i)}) \prod_{j=t+2}^{t+p} p(x_j \mid x_{j-1}) dx_{t+1:t+p-1} \end{aligned} \tag{4.01}$$

The evaluation of these integrals, though, may be difficult and/or may require significant computational effort, even in the case when a PF algorithm is used to approximate the state pdf for subsequent time instants. To simplify and solve this problem, three main approaches are now presented and explained in detail.

4.2.1. ***First Approach for Long-term Predictions: Weight Update Procedure***

This first approach predicts the evolution in time of each particle by successively taking the expectation of the model update equation (2.01) for every future time instant, and considering the state value associated to that particle as the initial condition, as shown in (4.02).

$$\hat{x}_{t+p}^{(i)} = E[f_{t+p}(\tilde{x}_{t+p-1}^{(i)}, \omega_{t+p})] \quad ; \quad \hat{x}_{t}^{(i)} = \tilde{x}_{t}^{(i)} \tag{4.02}$$

The weight of every particle should be modified (at each prediction step) to take into account the fact that the noise and process nonlinearities could change the shape of the state pdf as time passes. However, since the weight update procedure is needed as part of a prediction problem, it cannot depend on the acquisition of new measurements. To solve this difficulty, a procedure based on the use of the process noise model has been developed and implemented.

The procedure is as follows: consider the predicted conditional pdf $\hat{p}(x_{t+k}^{(i)} \mid \hat{x}_{t+k-1}^{(i)})$, which describes the state distribution at the future time instant $t+k$ $(k=1,\cdots,p)$ when the particle $\hat{x}_{t+k-1}^{(i)}$ is used as initial condition. Assuming that the initial weights $\left\{w_t^{(i)}\right\}_{i=1}^{N}$ are a good representation for the current state pdf, then it is possible to approximate the predicted state pdf at time $t+k$ by using the law of total probabilities and the particle weights at time $t+k-1$, as shown in (4.03):

$$\hat{p}(x_{t+k} \mid \hat{x}_{1:t+k-1}) \approx \sum_{i=1}^{N} w_{t+k-1}^{(i)} \cdot \hat{p}(x_{t+k}^{(i)} \mid \hat{x}_{t+k-1}^{(i)}) \; ; \; k = 1, \cdots, p \, . \tag{4.03}$$

As it was mentioned before, (4.03) assumes that the updated weights $w_{t+k-1}^{(i)}$ represent an accurate sampled version for $\hat{p}(x_{t+k-1} \mid x_{0:t}, y_{1:t})$, namely the predicted state pdf for the previous time instant. If in addition the domain of the particle population $\left\{\hat{x}_{t+k}^{(i)}\right\}_{i=1}^{N}$ is considered to be a representative subset for the domain for the state random variable x_{t+k}, then the following algorithm may be used to assign adequate values to each particle weight at the prediction time $t+k$:

Weight Update for Long-Term Prediction

- Construct a partition of the random variable domain by defining:

$$d_{t+k}^{(1)} = -\infty; \quad d_{t+k}^{(N+1)} = \infty$$
$$d_{t+k}^{(j)} = \frac{1}{2}\left(\hat{x}_{t+k}^{(j)} + \hat{x}_{t+k}^{(j-1)}\right), \quad j = 2, \ldots, N$$

- Generate the updated particle weights by computing:

$$w_{t+k}^{(i)} = \int_{d_{t+k}^{(i)}}^{d_{t+k}^{(i+1)}} \hat{p}(x_{t+k} \mid \hat{x}_{0:t+k-1}, y_{1:t}) dx_{t+k}$$

The proposed method is easy to implement as long as the process noise is uncorrelated (diagonal covariance matrix for $\omega(t)$). If that assumption does not hold, though, the integral associated with the weight update step may not be solvable.

4.2.2. ***Second Approach for Long-term Predictions: Regularization of Predicted State Probability Density Function***

The second approach for long-term prediction intends to avoid the computational effort implied in the update of the particle weights for future time instants, especially if the prediction time horizon is large. In this sense, instead of recalculating the particle weights, uncertainty for future transitions is incorporated by simply resampling the predicted state pdf (4.03).

Thus, the information about the distribution of the state for future time instants is now given by the position of the particles, not by the particle weight value. The implementation of this methodology, however, must ensure that the resampled population is representative of (4.03). A computationally affordable solution for this predicament is proposed, based on the assumption of uncorrelated process noise (diagonal covariance matrix for $\omega(t)$) and the use of kernel transitions to describe the state pdf before the resampling step, as it is also done in the case of the regularized particle filter (RPF).

Consider, in this sense, a discrete approximation (4.04) for the predicted state pdf (4.03), where $K(\cdot)$ is a kernel density function, which may correspond to the process noise pdf, a Gaussian kernel or a rescaled version of the Epanechnikov kernel (2.32) [24].

$$\hat{p}(x_{t+k} \mid \hat{x}_{1:t+k-1}) \approx \sum_{i=1}^{N} w_{t+k-1}^{(i)} K\left(x_{t+k} - E\left[x_{t+k}^{(i)} \mid \hat{x}_{t+k-1}^{(i)}\right]\right) \tag{4.04}$$

It is reasonable to try to represent the uncertainty present in (4.04), instead of just projecting the conditional expectations of the state variables. One way to achieve this task is to generate a new population of equally weighted particles for the time instant $t+k$, $1 \le k \le p$, performing an inverse transform resampling [41] procedure for the particle population. This method obtains samples distributing according to (4.04), selecting N realizations of $u^{(i)} \sim U(0,1)$ and interpolating a value for $\hat{x}_{t+k}^{(i)}$ from the cumulative state distribution $F(X_{t+k} \le x_{t+k}) = \int_{-\infty}^{x_{t+k}} \hat{p}(x_{t+k} \mid \hat{x}_{1:t+k-1})\, dx_{t+k}$ in accordance with $\hat{x}_{t+k}^{(i)} = F^{-1}(u^{(i)})$.

The inherent randomness present in the inverse transform resampling method, however, may lead to unrepresented areas in the domain of the cumulative state distribution function, a situation which is difficult to correct in long term predictions since there are no measurements available that may be used for this purpose. To overcome this difficulty, a two-step procedure is proposed.

The first step in the resampling strategy performs a simplified version of the inverse transform resampling procedure, which will focus on representing the growth of uncertainty present in (4.04). Samples distributing according to (4.04) are obtained by selecting $u^{(i)} = \dfrac{i}{N+1}$ $(i:1,\cdots,N)$, and interpolating a value for $\hat{x}_{t+k}^{(i)}$ from the cumulative state distribution $F(X_{t+k} \le x_{t+k}) = \int_{-\infty}^{x_{t+k}} \hat{p}(x_{t+k} \mid \hat{x}_{1:t+k-1})\, dx_{t+k}$ in accordance with $\hat{x}_{t+k}^{(i)} = F^{-1}(u^{(i)})$.

To avoid loss of diversity among particles, an additional step inspired by the RPF is performed. In this sense, it is assumed that the state covariance matrix $\hat{S}_{t+k}$ is equal to the empirical covariance matrix of $\hat{x}_{t+k}$ and that a set of equally weighted samples for $\hat{x}_{t+k-1}$ is available, in such a way that the efficiency in the use of Epanechnikov kernels for pdf approximation is maximized [24].

In consequence, considering all of the above, the regularization algorithm applied for long term predictions is as follows:

Long Term Predictions: Regularization of Predicted State PDF

- Apply modified inverse transform resampling procedure. For $i = 1, \cdots, N$, $w_{t+k}^{(i)} = N^{-1}$
- Calculate $\hat{S}_{t+k}$, the empirical covariance matrix of $\left\{ E\left[x_{t+k}^{(i)} \mid \hat{x}_{t+k-1}^{(i)} \right], w_{t+k}^{(i)} \right\}_{i=1}^{N}$
- Compute $\hat{D}_{t+k}$ such that $\hat{D}_{t+k}\hat{D}_{t+k}^{T} = \hat{S}_{t+k}$
- For $i = 1, \cdots, N$, draw $\varepsilon^{i} \sim K$, the Epanechnikov kernel and assign

  $\hat{x}_{t+k}^{(i)*} = \hat{x}_{t+k}^{(i)} + h_{t+k}^{opt}\hat{D}_{t+k}\varepsilon^{i}$, where h_{t+k}^{opt} is computed as in (2.33)

It is important to notice that the assumption of uncorrelated process noise is only included for the sake of reducing the computational effort of the resampling procedure. In fact, there are no theoretical restrictions for the application of this methodology in the presence of correlated process noise.

4.2.3. Third Approach for Long-term Predictions: Projection in Time of State Expectations

The third and last approach for long-term prediction presented in this section is, in fact, simpler in terms of computational effort than the previous ones. Whereas the first approach defines an update equation for the particle weights and the second approach intends to apply resampling steps for future time instants, the third approach states that the error that can be generated by considering the particle weights invariant for future time instants is negligible with respect to other sources of error that may appear in practical applications, such as model inaccuracies or even in the assumptions made for process and measurement noise parameters [4].

Therefore – from this standpoint – (4.02) is considered sufficient to extend the trajectories $\hat{x}_{0:t+k}^{(i)}$, while the current particle weights are propagated in time without changes. The computational burden of this method is significantly smaller and, as it will be shown in simulation results, the method still offers a satisfactory view about how the system behaves for most practical applications.

4.3. Second Prognosis Level: Statistical Characterization of the Remaining Useful Life (RUL) for Pieces of Equipment

The final outcome for any prognosis algorithm is an estimate for the system RUL pdf, which is intrinsically related to the probability of failure at future time instants. This probability can be obtained from long-term predictions, when the empirical knowledge

about critical conditions for the system is included in the form of thresholds for main fault indicators, also referred to as the hazard zones.

In real applications, it is expected for the hazard zones to be statistically determined on the basis of historical failure data, defining a critical pdf with lower and upper bounds for the fault indicator (H_{lb} and H_{ub}, respectively).

Since the hazard zone specifies the probability of failure for a fixed value of the fault indicator, and the weights $\left\{w_{t+k}^{(i)}\right\}_{i=1}^{N}$ represent the predicted probability for the set of predicted paths, then it is possible to compute the probability of failure at any future time instant (namely the RUL pdf) by applying the law of total probabilities, as shown in (4.05). Once the RUL pdf is computed, combining the weights of predicted trajectories with hazard zone specifications, it is well known how to obtain prognosis confidence intervals, as well as the RUL expectation.

$$\hat{p}_{TTF}(ttf) = \sum_{i=1}^{N} \Pr\left(Failure \mid X = \hat{x}_{ttf}^{(i)}, H_{lb}, H_{ub}\right) \cdot w_{ttf}^{(i)} \qquad (4.05)$$

Expression (4.05) provides a solution for the RUL pdf estimation problem that is very suitable for on-line applications. Since it depends on the predicted trajectory weights, though, it is subject to uncertainty and it may be sensitive to modeling errors. Moreover, uncertainty inherent to RUL expectations increases as the prediction horizon grows. This issue is of especial interest in prognosis, since the estimation of the RUL

must be done immediately after the fault condition has been detected, and hence most of the prediction horizons involve considerably long time periods.

Due to this reason, the present thesis has included the implementation of *outer correction loops* as part of the second prognosis level. These *correction loops* intend to update parameters of great significance in the overall performance of FDI and/or prognostic algorithms. The following subsections focus on two particular *correction loops*: (1) an autoregressive correction algorithm utilized to improve accuracy in RUL expectations, and (2) a model parameter update procedure that facilitates identification of nonlinear systems undergoing changes in operational conditions.

4.3.1. Outer Correction Loop in Failure Prognosis: Accuracy in RUL Estimate

The first *outer correction loop* that has been included in this work intends to improve the accuracy of the RUL expectation; see Figure 4.1. This *outer loop* is basically a data-driven learning paradigm that intends to correct inaccuracies in the nonlinear dynamic model used in the prognostic module. For this purpose, it considers the effects that the previous measurements had in the RUL estimates, under the assumption that the expected value for the sum of the effects is null when the model truly represents the evolution of the fault condition in time.

The algorithm computes a series of correction terms C_j ($j = 1,\ldots,k$) that measure the difference between the RUL expectation computed at the current time $t = j$ and the one that was computed in the previous iteration of the prognosis algorithm (i.e.

the RUL expectation computed at $t = j-1$). Once k correction terms are computed, a linear autoregressive model is built to establish a relationship between $C_1,\ldots,C_k$.

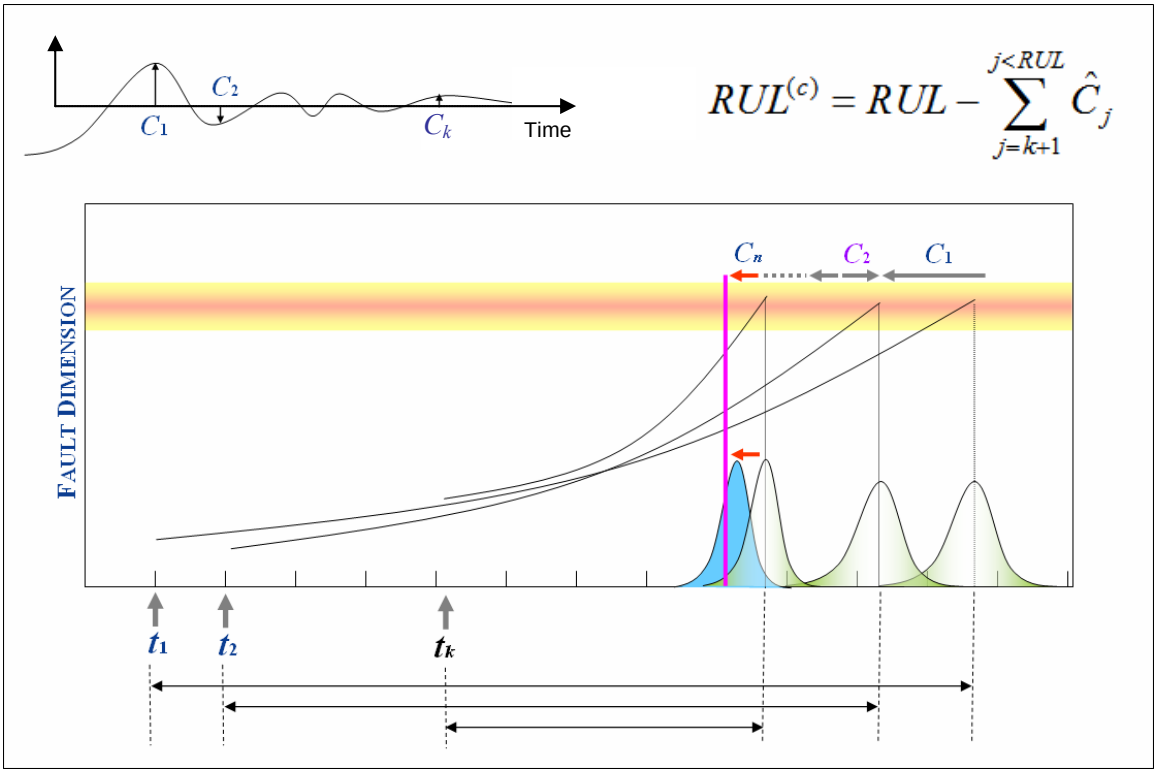

Figure 4.1. Outer correction loop for RUL expectation. The algorithm considers the differences between RUL expectations computed at different time instants. A regression model is then built to estimate C_n, a quantity representing the consistency of the prediction results, which modifies the final RUL estimate

The obtained linear autoregressive model is then used to estimate all future corrections $\hat{C}_{k+1},\ldots,\hat{C}_{RUL}$ that would be applied to the current RUL expectation if measurement data were to be acquired until the failure time, assuming that both process and measurement noises are wide sense stationary (WSS). Finally, the current RUL

expectation is modified as shown in (4.06) to obtain $RUL^{(c)}$, the corrected estimate for the remaining useful life at time $t = k$.

$$RUL^{(c)} = RUL - C_n$$
$$C_n = \sum_{j=k+1}^{j<RUL} C_j \qquad (4.06)$$

In simple words, the proposed *outer correction loop* intends to capture the pattern of past measurement-driven prediction updates inside a simple model, providing a measure for the consistency of the prediction results, which can be used afterwards to estimate and correct for the accuracy of the current prediction.

The learning scheme proposed here is an example of how the combination of model-based (particle-filtering) and data driven (linear autoregressive correction models) techniques in an *outer correction loop* can significantly improve the prognosis algorithm accuracy.

4.3.2. Outer Correction Loop in Failure Prognosis: Model Parameter Adjustment

The second proposed *outer correction loop* intends to facilitate the adaptability of the nonlinear dynamic model that has been used for prognostic purposes in the event of drastic changes in the operating conditions of the system. In particular, this loop uses feedback concepts to modify parameters that define the profile of the process noise used in the aforementioned nonlinear dynamic model.

Let the dynamic nonlinear model (4.07) be used for purposes of predicting the evolution of a fault indicator in time:

$$\begin{cases} x(t+1) = f_t(x(t), x_\alpha(t), \omega_1(t)) \\ x_\alpha(t) = x_\alpha(t) + \omega_\alpha(t) \\ \text{Features}(t) = h_t(x(t), x_\alpha(t), v(t)) \end{cases}, \qquad (4.07)$$

where x is the state vector, f_t and h_t are nonlinear mappings, ω_1 and v are non-Gaussian noises, ω_α is a zero mean random noise, and $x_\alpha(t)$ is a state associated with an unknown model parameter α .

Furthermore, let the unknown parameter α be fixed under typical operating conditions. As it was previously discussed in Section 2.4.3, the concept of "artificial evolution" [16] may be used to estimate the value of the model parameter, with the help of kernel smoothing methods. However, the problem with this method arises when the monitored system undergoes a sudden change in the operating condition, or when the model parameter unexpectedly changes, since the assumption of a constant parameter does not hold any longer.

This problem may be solved using a *correction loop* that utilizes the short-term prediction error to modify the variance of ω_α, which is the noise signal that is used to estimate the model parameter α through a particle-filter algorithm and the model (4.07). Several approaches may be proposed to implement a *correction loop* of these characteristics. In this sense, a major aspect that has to be considered is the expected

frequency and severity of the changes in the operating conditions of the system. In particular, this thesis proposes a *correction loop* based on the expression (4.08):

$$\begin{cases} \text{var}\{\omega_\alpha(t+1)\} = p \bullet \text{var}\{\omega_\alpha(t)\}, \text{ if } \dfrac{\|Pred\_error(t)\|}{\|Feature(t)\|} < Th \\ \text{var}\{\omega_\alpha(t+1)\} = q \bullet \text{var}\{\omega_\alpha(t)\}, \text{ if } \dfrac{\|Pred\_error(t)\|}{\|Feature(t)\|} > Th \end{cases}, \tag{4.08}$$

where *Pred_error*(*t*) is the short-term prediction error computed at time *t* (for example, 5-step prediction error), $\|\cdot\|$ is any well-defined norm (usually L_2-norm), $0 < p < 1$, $q > 1$, and $0 < Th < 1$ are scalars. In particular, the author recommends using $p \in [0.925, 0.975]$, $q \in [1.10, 1.20]$, and $Th = 0.1$. Recommended values have been determined through exhaustive analysis of simulations considering scenarios with different combination of values for the parameters *p*, *q*, and *Th*.

This approach allows the value of the unknown model parameter to be rapidly updated if the prediction error becomes large due to the fact that the model no longer represents the dynamics of the faulty system. On the other hand, if the prediction error is consistently small, the variance of the state $x_\alpha(t)$, associated with the model parameter, decreases exponentially, thus helping to stabilize the structure of the model and the corresponding RUL estimate. A detailed performance analysis for this proposed *correction loop* is provided in Section 5.2.2.2, where the algorithm is tested for two different scenarios: (1) erroneous initial condition for the model parameter and (2) sudden change in the value of model parameter.

5. PROGNOSIS IN NONLINEAR SYSTEMS: CASE STUDIES

5.1. Scope and Aim of the Chapter

Several approaches that can be considered to solve the problem of RUL statistical estimation, using a particle-filtering-based methodology, have been introduced in Chapter 4. Whereas some of them are intended to generate a more accurate representation of the RUL probability density function, others aim to simplify the computational burden so that on-line applications can be easily implemented. This chapter provides several case studies in order to analyze and properly understand the main advantages and disadvantages that these methods may offer. In all these case studies, the implementation of prognostic algorithms considered 20 particles for purposes of pdf state estimation.

5.2. Illustrative Example: RUL Statistical Characterization

5.2.1. Evaluation of Prognostic Approaches and First Outer Correction Loop

Consider the problem of RUL estimation in a process for which the evolution in time of a known failure condition (for instance, a crack in a material) is described by the nonlinear system (5.01), where the state $x_1(t)$ is associated with the fault dimension that is being analyzed, the state $x_2(t)$ represents a time-varying model parameter that directly affects the rate of growth of the fault dimension, $y(t)$ represents the measurements, and $\omega_2(t)$ is zero mean Gaussian noise. Model (5.01) is needed to generate the data that is used for purposes of evaluating the proposed prognostic routines.

$$\begin{cases} x_1(t+1) = x_1(t) + 3\cdot 10^{-4}\left(0.05 + 0.1\cdot x_2(t)\right)^3 + \omega_1(t) \\ x_2(t+1) = x_2(t) + \omega_2(t) \\ y(t) = x_1(t) + v(t) \end{cases} \tag{5.01}$$

$$\omega_1(t) \sim Gamma(0.15, 0.3)$$

$$v(t) \sim \frac{1}{4}\mathcal{N}(-0.5, 0.25) + \frac{3}{4}\mathcal{N}(0.5, 0.25)$$

To analyze the effect that inaccuracies and model errors imply in RUL estimates, noise profiles in system (5.01) are assumed to be Gaussian in the prognosis model. The first two moments of both the process and observation noise in this prognosis model may be estimated using historical data (generated by using model (5.01)). As a result, the particle-filter-based framework for prognosis is finally built upon the nonlinear dynamic model (5.02), where $\omega_2(t)$ is zero mean Gaussian noise.

$$\begin{cases} x_1(t+1) = x_1(t) + 3\cdot 10^{-4}\left(0.05 + 0.1\cdot x_2(t)\right)^3 + \omega_1(t) \\ x_2(t+1) = x_2(t) + \omega_2(t) \\ y(t) = x_1(t) + v(t) \end{cases} \tag{5.02}$$

$$\omega_1(t) \sim \mathcal{N}(0.045, 0.1162)$$

$$v(t) \sim \mathcal{N}(0.25, 0.5)$$

The hazard zone, which in real applications must be defined on the basis of customer specifications or ground truth failure data, is defined here as a normal probability density function with parameters $\mu = 9.0$ and $\sigma = 0.3$. The main objective is to generate a 95% confidence interval for the RUL of the process, 40 cycles after the fault condition is detected.

In addition to the techniques described in Chapter 4, an Extended Kalman Filter (EKF)–based prognosis procedure has also been considered as a means for both comparison and performance evaluation for the proposed particle-filtering-based techniques.

Figure 5.1 shows the performance of the PF-based estimation algorithm in the case of the aforementioned problem. The green and magenta lines represent, respectively, the noisy measurements and the estimate of the process output obtained from a SIR particle-filter. The blue line shows the actual evolution of the fault condition in time.

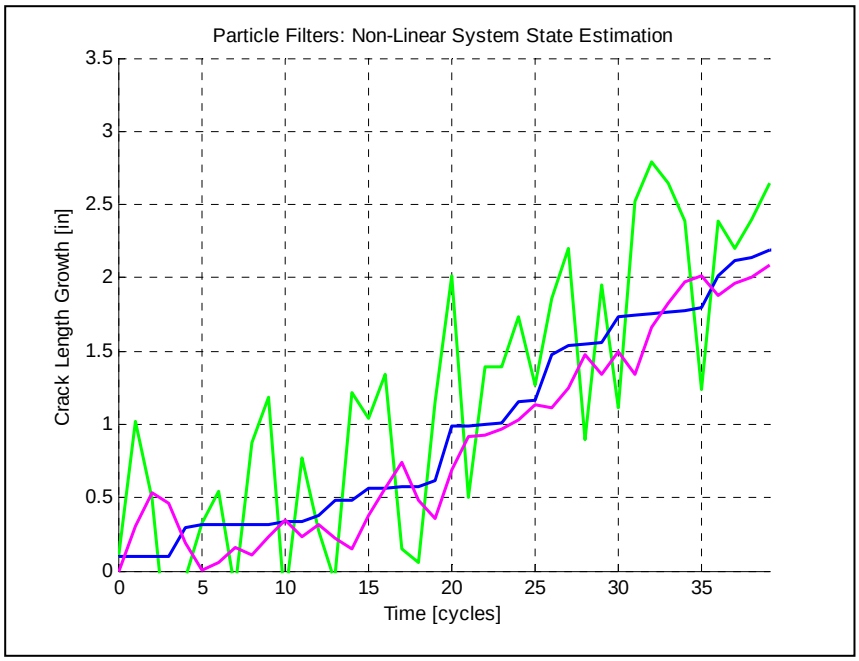

Figure 5.1. Result PF-based state estimation. The blue line represents the true value of the state $x_1(t)$. The green line represents the measurements and the magenta line is the state estimate.

Figure 5.2 summarizes the prognosis results obtained when applying every one of the particle-filter-based approaches described in Sections 4.1 and 4.2, and the *outer*

correction loop explained in Section 4.3.1. The green and magenta lines represent, respectively, the noisy measurements and the estimate of the process output obtained from a SIR particle-filter. The blue line shows the actual evolution of the fault condition for future time instants (information that is unknown at the moment where the RUL estimation is performed).

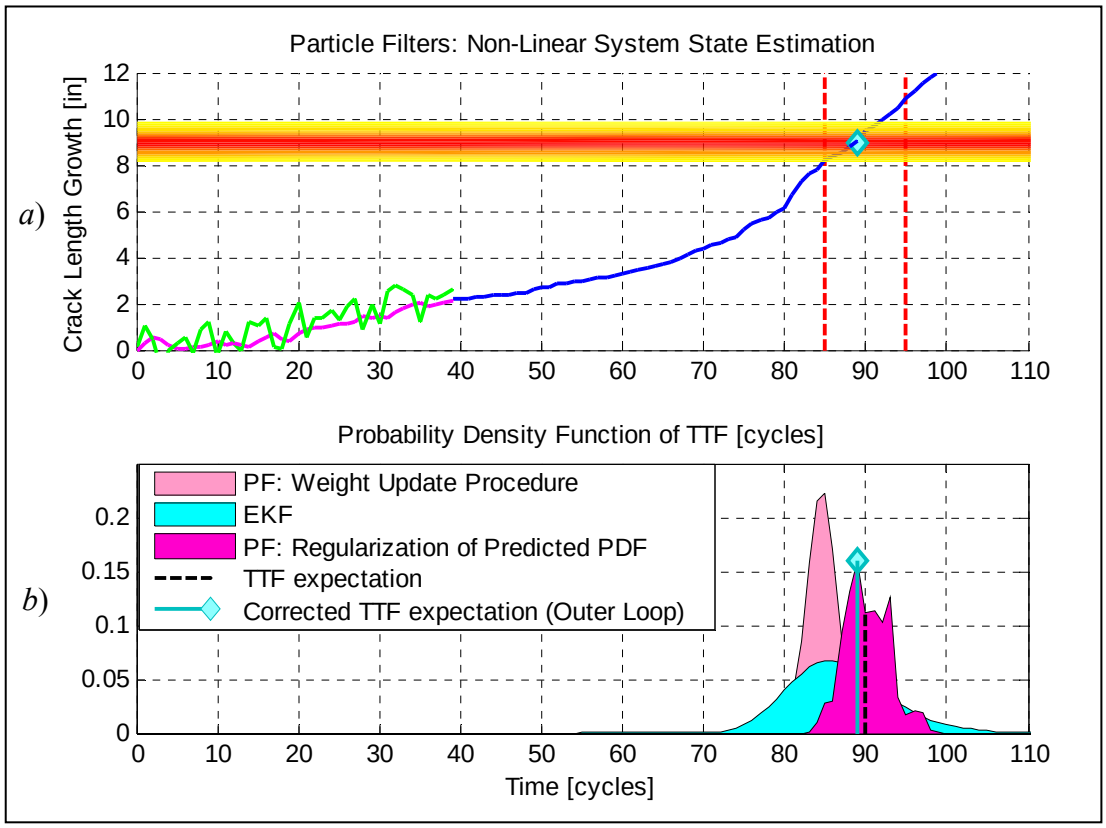

Figure 5.2. Result comparison for RUL statistical characterization. (a) Green line is the noisy measurements, magenta line is the estimate of the process output, blue line is the actual evolution of the fault condition, and orange area represents the hazard zone. (b) Pdf estimates for the RUL of the faulty system computed by the proposed particle-filter-based prognosis framework vs. EKF, the vertical line in cyan marks the output of the outer correction loop for the RUL estimate

The vertical red lines in Figure 5.2 (*a*) correspond to the upper and lower bounds of the RUL 95% confidence interval, respectively. This confidence interval is calculated by considering the estimate of the RUL pdf obtained from the application of the second approach for long-term predictions (regularization of the predicted state pdf).

Figure 5.2 provides helpful information to evaluate the capability of each algorithm for predicting the evolution in time of the state probability distribution, particularly when some performance metrics (such as precision and accuracy) are invoked to assess the algorithm performance.

For instance, consider the RUL (also referred to as TTF) pdf estimates obtained by using the PF–based first approach for long term predictions (weight update procedure) and the ones obtained with the EKF–based procedure. It is possible to notice an important difference in terms of the uncertainty of the prediction (precision), even though both techniques have the same expectation for the RUL of the process (accuracy). Differences in precision are mainly caused by the fact that the EKF assumes a normal pdf for the state, regardless of the likelihood of the estimation, whereas the PF is able to discard realizations of the state if the probability masses associated with them are negligible.

The major problem with both techniques, however, seems to be related to the accuracy of the algorithm, i.e., the capability of estimating the expectation of the RUL for the process. In the case of the EKF–based procedure, this issue is related to the fact that long term predictions greatly depend on the current expectation of the state, an assumption that may not be representative of the non-linear process behavior in a long

term horizon when model errors are included. Similar problems affect the PF–based first approach for long term predictions (weight update procedure), since the algorithm depends strongly on those particles which have a higher likelihood, given the current observation data.

On the other hand, as it is also shown in Figure 5.2, the PF–based second approach for long term predictions (regularization of the predicted state pdf) is capable of overcoming the bias introduced by model errors, due to its ability to represent the state probability space. The combination of resampling techniques and Epanechnikov kernels for pdf approximation in long term predictions is able to simultaneously reduce the impact of model inaccuracies and provide a balanced result in terms of accuracy and precision in the RUL estimate. Furthermore, the actual fault indicator (unknown when the long term predictions were performed) reaches the previously defined hazard zone inside the 95% confidence interval, confirming the validity of the RUL pdf estimate.

Results for the PF-based third approach for long term predictions (projection in time of state expectations) are not shown in Figure 5.2 since they are quite similar to the ones obtained with the first approach. It is important to note, however, that the computational burden is considerably less in the case of the third approach and, thus it seems to be more suitable than the first one for on-line applications.

Finally, it must be noted that when the *outer loop* correction scheme ─ introduced as part of the second prognosis level ─ is applied to the PF-based second approach for long term prediction generation, it allows improvement of the estimate of the RUL

expectation to the extent that the corrected estimate of time-to-failure (TTF) coincides with the time instant where the actual failure growth reaches the mean value of the hazard zone.

5.2.2. Analysis of Second Outer Correction Loop in Failure Prognosis

Consider the problem of RUL estimation in a process for which the evolution in time of a known failure condition (for instance, a crack in a material) is described by the nonlinear system (5.03), where $\omega_2(t)$ is zero mean Gaussian noise and γ is a fixed model parameter.

$$\begin{cases} x_1(t+1) = x_1(t) + \gamma \cdot 3 \cdot 10^{-4} \left(0.05 + 0.0325 \cdot x_2(t)\right)^3 + \omega_1(t) \\ x_2(t+1) = x_2(t) + \omega_2(t) \end{cases}$$
$$y(t) = x_1(t) + v(t) \qquad (5.03)$$
$$\omega_1(t) \sim \mathcal{N}(0.045, 0.1162)$$
$$v(t) \sim \mathcal{N}(0.25, 0.5)$$

It is important to note that model (5.03) differs from model (5.01) regarding the definition of the process and measurement noise distributions. Also one of the coefficients that affect the growth rate of the fault dimension has been decreased in model (5.03). The new model not only allows to simulate changes in the model parameter before the definite prognosis results are generated, but also helps to focus the analysis in the quality of the estimate of the model parameter rather than in the effect of model inaccuracies.

Similarly to what has been done in Section 5.2.1, the noise profiles in system (5.03) are assumed to be Gaussian. As a result, the particle-filter-based framework for prognosis is built, in this case, upon the nonlinear dynamic model (5.04), where $\omega_2(t)$ and $\omega_3(t)$ are zero mean Gaussian noise.

$$\begin{cases} x_1(t+1) = x_1(t) + 3\cdot 10^{-4}\cdot x_3(t)\cdot\left(0.05 + 0.0325\cdot x_2(t)\right)^3 + \omega_1(t) \\ x_2(t+1) = x_2(t) + \omega_2(t) \\ x_3(t+1) = x_3(t) + \omega_3(t) \end{cases}$$
$$y(t) = x_1(t) + v(t) \qquad (5.04)$$
$$\omega_1(t) \sim \mathcal{N}(0.045, 0.1162)$$
$$v(t) \sim \mathcal{N}(0.25, 0.5)$$

The main focus in this section is to evaluate the efficacy of the *outer correction loop* proposed in Section 4.3.2, in terms of the influence that this approach has in the overall performance of particle-filter-based estimation and prognostic procedures. For this purpose, two different scenarios involving difficulties in the estimation of model parameter γ – in model (5.03) – have been considered: (1) Erroneous initial condition for the model parameter, and (2) sudden changes in the value of the model parameter.

5.2.2.1. First scenario: Model Parameter Estimation with Erroneous Initial Condition

In this first scenario, the model parameter $\gamma = 1.5$ is assumed constant throughout the whole experiment. The initial condition for the state $x_3(t)$ – the state in model (5.04) associated with the estimate of the parameter γ – is set as $x_3(0) = 2.5$, which is 66%

higher that the actual value. It is of interest to analyze how an *outer correction loop*, based on the approach described in Section 4.3.2, helps to accelerate the convergence of the parameter estimate.

Given that model (5.04) is not completely observable, and thus there is no unique solution to the estimation problem, the initial conditions of states $x_1(t)$ and $x_2(t)$ have been set close to their actual values at time $t = 0$ to ensure an adequate comparison for the estimate of the state $x_3(t)$.

A 5-step prediction error has been used in the design of the *outer correction loop*. As expected, the longer the period considered to calculate the prediction error, the larger the delay in the feedback loop. Several aspects must be considered in a proper selection of this parameter – as well as for p, q and Th in equation (4.08) – including the time constants of the system and the variability of model parameters. The final implementation of the *correction loop* is shown in (5.05), while Figure 5.3 shows the obtained results:

$$\begin{cases} \text{var}\{\omega_3(t+1)\} = 0.95 \bullet \text{var}\{\omega_3(t)\}, \text{ if } \dfrac{\|Pred\_error(t)\|}{\|y(t)\|} < 0.1 \\ \text{var}\{\omega_3(t+1)\} = 1.20 \bullet \text{var}\{\omega_3(t)\}, \text{ if } \dfrac{\|Pred\_error(t)\|}{\|y(t)\|} > 0.1 \end{cases} . \qquad (5.05)$$

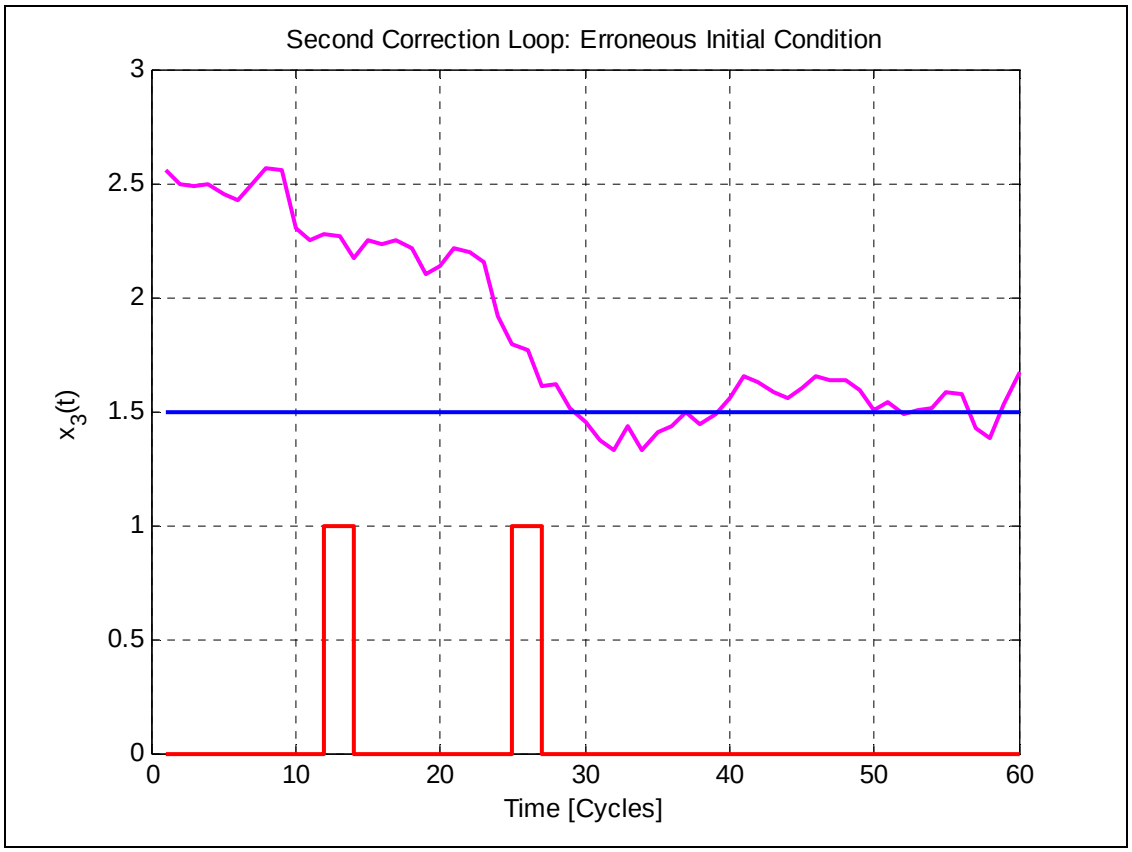

Figure 5.3. Blue line depicts the actual value of the unknown model parameter, magenta line is the particle-filter-based estimate of the $x_3(t)$ – state associated with that parameter –, and red line marks the time instants when the outer correction loop modifies the variance of $x_3(t)$ in the dynamic model

Figure 5.3 clearly shows two periods of time where the *outer correction loop* actually increments the variance of the noise profile $\omega_3(t)$ in the dynamic model (5.04). Especially after the second period, it is observed that the state estimate (blue curve) rapidly converges to the true value of the model parameter (blue horizontal line in Figure 5.3). After that condition is reached, the prediction error decreases considerably and therefore, the variance of the noise used for the "artificial evolution" estimation method [16] starts decreasing exponentially.

Although the variance of the noise profile $\omega_3(t)$ decreases in time, this fact does not necessarily mean that the state estimate will converge in the same manner. This phenomenon is particularly true in a particle-filtering algorithm, where the weights are individually affected by the likelihood of the measurements. In cases where the value of the model parameter frequently changes, it is also critical to use a proper prediction horizon to ensure convergence and avoid unnecessary disturbances in the filtering scheme. Nevertheless, as the proposed *outer correction loop* modifies only parameters of the *prior* distribution, effects of delays in the feedback loop are diminished by the decrement in the weights of particles with low likelihood. Further studies about convergence and stability issues are recommended as part of the future work.

Lastly, Figure 5.4 depicts the results obtained for the computation of the RUL pdf estimate at time $t = 60$, using the particle-filtering-based approach discussed in Section 4.2.2. Given that the initial conditions for the states $x_1(t)$ and $x_2(t)$ are close to the actual values of those states at time $t = 0$, and that the estimate of state $x_3(t)$ converged to the true value of the model parameter γ, it is expected for the mean value of the RUL pdf to be close to the actual time-to-failure.

In general, from the results presented in Figure 5.3 and Figure 5.4, the implemented *outer correction loop* accomplishes the objective of contributing to the overall performance of the prognosis framework. Next section focuses the performance of the algorithm in the event of sudden changes in the value of the model parameter.

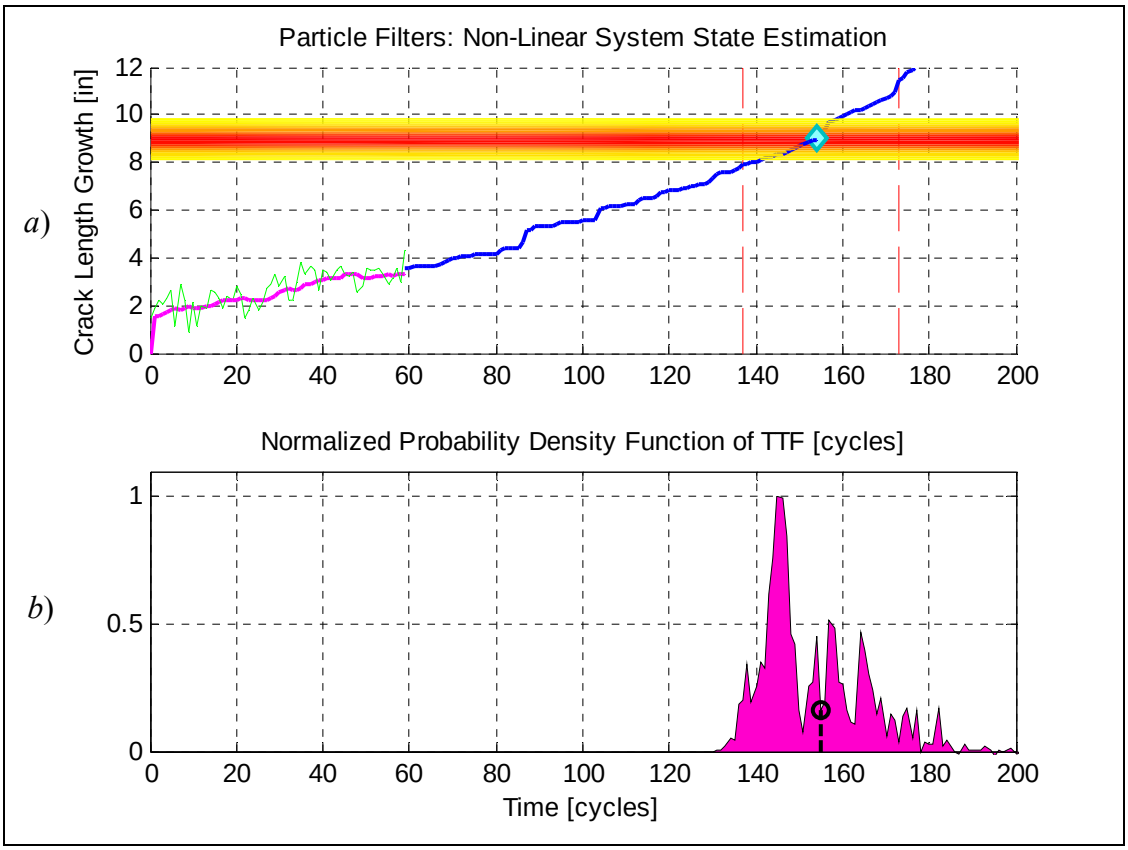

Figure 5.4. (a) Green line is the noisy measurements, magenta line is the estimate of the process output, blue line is the actual evolution of the fault condition, and orange area represents the hazard zone. (b) Pdf estimate for the RUL of the faulty system computed by the proposed particle-filter-based prognosis framework, the black vertical line marks the RUL expectation

5.2.2.2. Second Scenario: Model Parameter Estimation in the Event of Changes in Operating Conditions

The second scenario in the analysis considers the case where the model parameter γ in model (5.03) undergoes a sudden increment of 66% of its value at time $t = 30$, i.e.

$$\begin{cases} \gamma = 1.5, \text{ if } t < 30 \\ \gamma = 2.5, \text{ if } t \geq 30 \end{cases}. \tag{5.06}$$

The initial condition for the state $x_3(t)$ – the state in model (5.04) associated with the estimate of the parameter γ – is set as $x_3(0) = 1.0$, which is 33% smaller that the actual value at that time instant. It is of interest to analyze how an *outer correction loop*, based on the approach described in Section 4.3.2, helps to accelerate the convergence of the parameter estimate under the stated conditions.

Given that model (5.04) is not completely observable, and thus there is no unique solution to the estimation problem, the initial conditions of states $x_1(t)$ and $x_2(t)$ were set close to their actual values at time $t = 0$ to ensure an adequate comparison for the estimate of the state $x_3(t)$.

A 5-step prediction error has been used in the design of the *outer correction loop*. As it has been mentioned in Section 5.2.2.1, the longer the period considered to calculate the prediction error, the larger the delay in the feedback loop. The implementation of the *correction loop* is shown in (5.05), while Figure 5.5 shows the obtained results.

Figure 5.5 shows several periods of time where the *outer correction loop* actually increments the variance of the noise profile $\omega_3(t)$ in the dynamic model (5.04). The first one of them intends to help the convergence of the parameter in similar manner as in Section 5.2.2.1. In fact, comparing the first cycles of Figure 5.3 and Figure 5.5 it can be note that the correction loop in the latter case reacts later in time, since the difference between the actual parameter value and the initial condition is not as strong as in the case of Section 5.2.2.1 (and thus, the prediction error is not that significant).

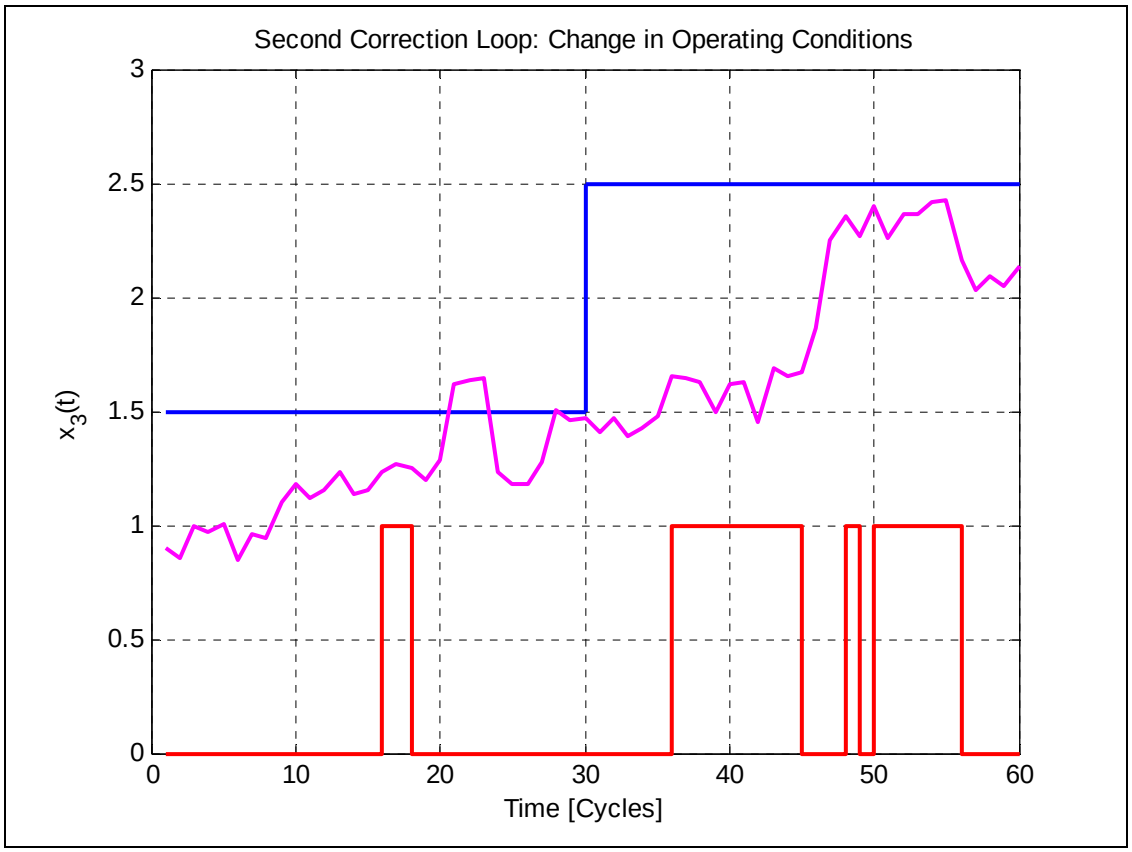

Figure 5.5. Blue line depicts the actual value of the unknown model parameter, magenta line is the particle-filter-based estimate of the $x_3(t)$ – state associated with that parameter –, and the red line marks the time instants when the outer correction loop modifies the variance of $x_3(t)$ in the dynamic model

After the sudden change in the parameter value, the difference between the estimate and its true value forces to increment the variance of $\omega_3(t)$ for a considerable amount of time. Although the state estimate (blue curve) rapidly starts to converge to the true value of the model parameter (blue horizontal line), more simulation time would be needed to achieve convergence. It is important to mention that the simulation time is restricted in this case to provide a significant prediction window for prognosis purposes,

and also to afford a fair basis of comparison with respect to the case described in Section 5.2.2.1.

As mentioned in the analysis of Section 5.2.2.1, the sustained increment in the variance of the noise profile $\omega_3(t)$ does not necessarily generate instability issues in the feedback loop. In fact, the overall performance of the particle-filter-based prognosis framework (see Figure 5.6) proved to be satisfactory.

Figure 5.6 depicts the results obtained for the computation of the RUL pdf estimate at time $t = 60$, using the particle-filtering-based approach discussed in Section 4.2.2. Given that the initial conditions for the states $x_1(t)$ and $x_2(t)$ are close to their actual values at time $t = 0$, and that the estimate of state $x_3(t)$ converged to a reasonably close value of the model parameter γ, would be reasonable to expect that the true value of the RUL would be within the 95% confidence interval for the time-to-failure. Although the change in the parameter value affected the accuracy of the prognosis algorithm, the *outer correction loop* helped to mitigate this effect by increasing the rate of convergence of the state estimate.

In general, from the results presented in Figure 5.5 and Figure 5.6, the implemented *outer correction loop* succeeds in providing a reasonable estimate of the unknown model parameter in a complex scenario involving both an erroneous initial condition and a sudden change in the value of the model parameter. Future work, as it is stated in Section 5.2.21, should consider stability issues for this type of feedback loops.

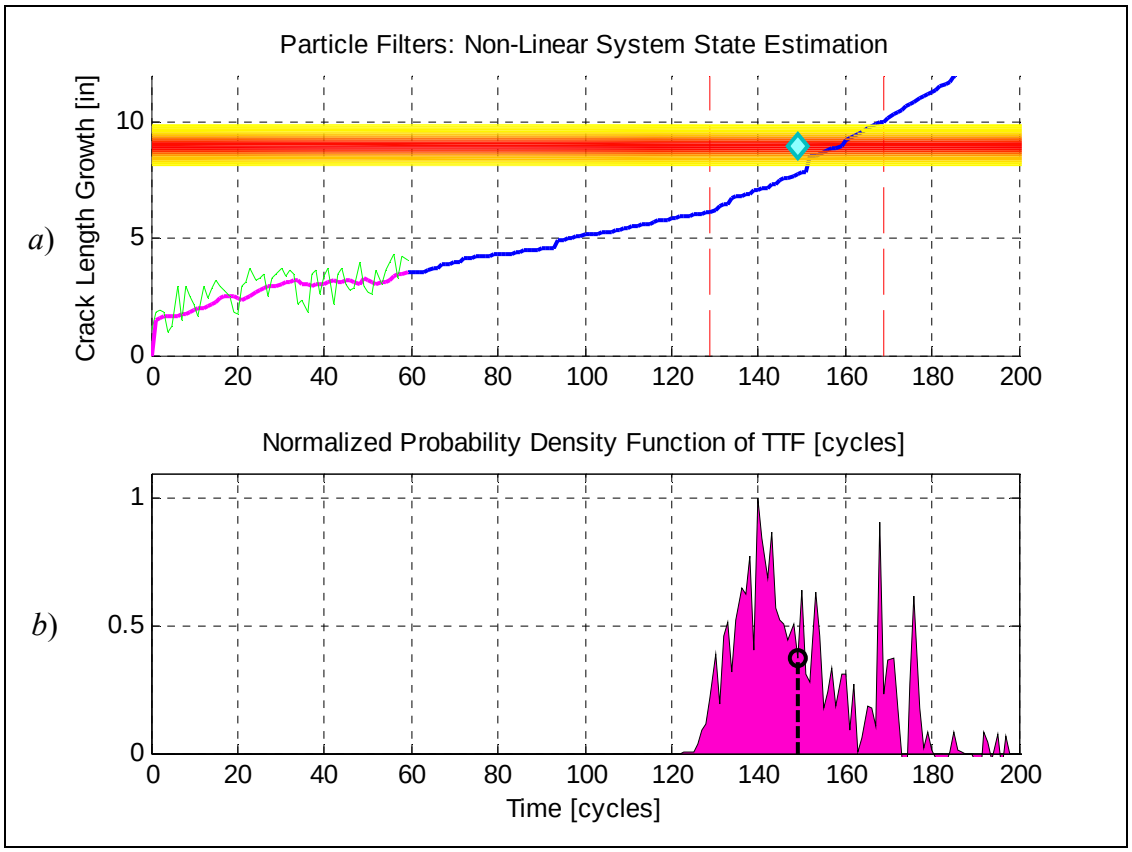

Figure 5.6. (a) Green line is the noisy measurements, magenta line is the estimate of the process output, blue line is the actual evolution of the fault condition, and orange area represents the hazard zone. (b) Pdf estimate for the RUL of the faulty system computed by the proposed particle-filter-based prognosis framework, the black vertical line marks the RUL expectation

5.3. Case Study: Analysis of Crack Growth in a Turbine Engine Blade

The implementation and testing of the proposed particle-filtering-based methodology for fault prognosis on real process data, and the subsequent assessment of the obtained results, are amongst the most important contributions intended for this thesis. With this purpose, two different sets of data have been used to define study cases related to the analysis of the growth of a crack in a rotatory piece of equipment.

For the first case study, consider the problem discussed in Section 3.2 where the objective is to analyze the growth in a crack on blades of a turbine HPC disk, see Figure 3.1. As was previously mentioned, light probes on both the leading and the trailing edge of the blades were installed to provide the time-of-arrival (TOA) for each blade, and this information was processed to come up with a feature directly related to the size of the crack in the blade, namely the tangential blade position (TBP). Although Section 3.2 was primarily focused in the use of this feature for FDI purposes, the obtained results may be also used to set up a failure prognosis framework where not only the size of the crack is estimated, but also a 95% confidence interval for the RUL of the piece of equipment may be computed.

With this purpose, as it was also mentioned in Section 3.2, an exhaustive turbine structural analysis was conducted in a FRANC-3D environment to determine a model capable of describing the growth of a crack in one blade under nominal load conditions. Results of this structural analysis are summarized in model (5.07) where L is the length of the crack (in inches), n is the number of stress cycles applied to the material, α is a random variable with known first two moments, $p(L(n))$ is a known fourth order polynomial determined with the help of FRANC-3D structural model, and $\omega(n)$ and $v(n)$ are independent and identically distributed (i.i.d.) white noises.

$$\begin{aligned} \frac{dL}{dn} &= \frac{1}{6\alpha \cdot L^5(n) + p(L(n))} + \omega(n) \\ y(n) &= h^{-1}(L(n)) + v(n) \end{aligned} \tag{5.07}$$

The measurement equation is considered to be linear on $h^{-1}(\cdot)$, where $h(\cdot)$ is a known nonlinear mapping between the crack size and feature value. Besides the fact that the state equation in non-linear, the noise signal $\omega(n)$ is non-negative, and thus non-Gaussian.

Model (5.07) is suitable for the generation of long term predictions and hence, for the implementation of a particle-filtering-based framework for failure prognosis. Moreover, a similar realization has been already used for fault detection purposes, giving evidence that one of the blades had a crack at the 280th cycle of operation. Thus, FDI results can be included as initial conditions for prognosis routines by assigning the resulting pdf estimate for the state $x_c(t)$ at $t = 280$, from model (3.04), as the initial particle population for the state $x_{c,1}(t)$ of the model (5.08).

$$\begin{cases} x_{c,1}(t+1) = \left(1 + \dfrac{1}{6 \cdot x_{c,2}(t) \cdot x_{c,1}^5(t) + p(x_c(t))}\right) x_{c,1}(t) + \omega_1(t) \\ x_{c,2}(t+1) = x_{c,2}(t) + \omega_2(t) \\ \begin{bmatrix} x_{c,1}(280) \\ x_{c,2}(280) \end{bmatrix} = \begin{bmatrix} E[x_c(280)] \\ E[\alpha] \end{bmatrix} \\ \mathrm{y}(t) = h^{-1}(x_{c,1}(t)) + v(t) \end{cases} \tag{5.08}$$

Preliminary studies about the value of the parameter α may be conducted for nominal load conditions to determine a proper value for the initial condition of the second state $x_{c,2}(t)$. A hazard zone around 0.3" was defined according to customer specifications.

Results obtained by the application of a SIR particle filter for state estimation, and the implementation of the third approach for long term prediction generation (projection in time of state expectations) are shown in Figure 5.7. Both the RUL pdf estimate – also referred to as the time-to-failure (TTF) pdf – and the long term prediction bounds have been computed considering only 40 cycles of data after the detection time instant and model (5.08). Also, a population of 20 particles has been utilized in the algorithm.

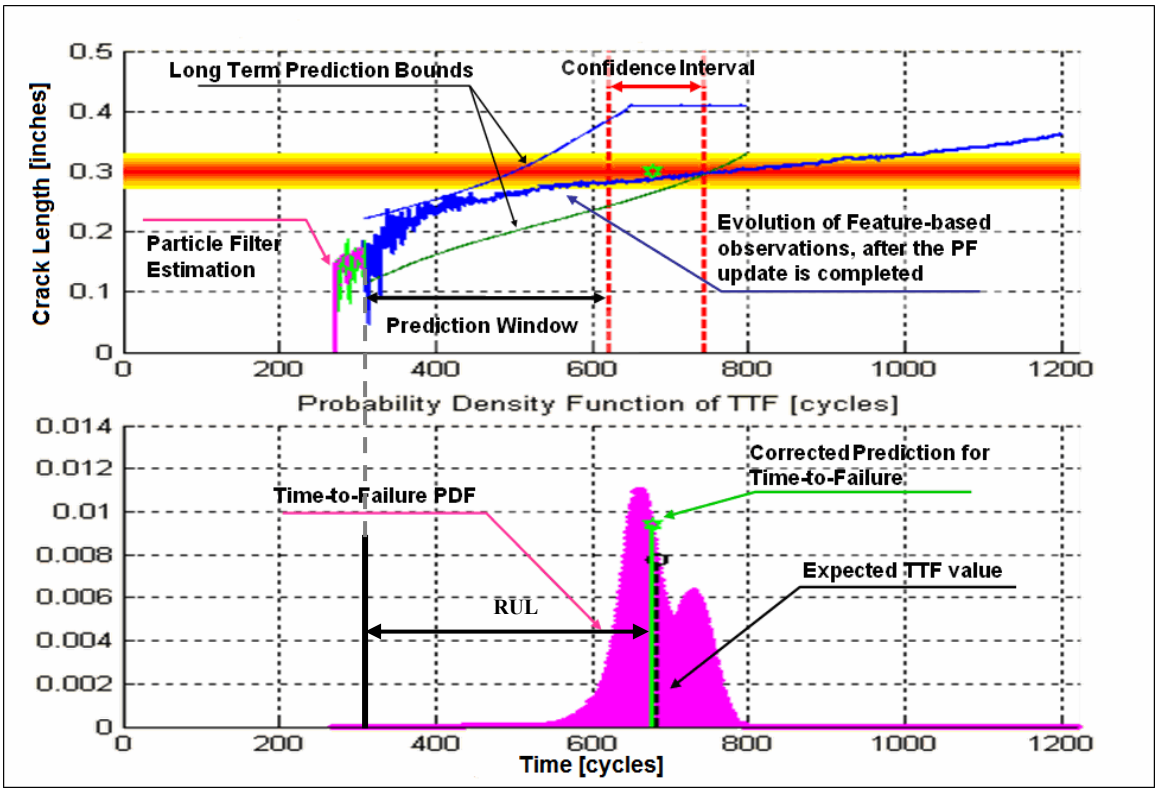

Figure 5.7. Prognosis results for crack growth in blades of a turbine HPC disk. (a) Long-term prediction bounds vs. actual fault data. The light green line represents the measurement data. The magenta line is the particle-filter-based state estimate and the blue line is the actual progression of the fault dimension after the prognosis results are generated. (b) RUL pdf estimate and depiction of prediction window used at the moment of generating the estimates

Obtained results are excellent in terms of accuracy for both the estimated expected failure time and its 80% confidence interval, offering also a prediction time window of approximately 300 cycles, which is reasonable for corrective actions before the crack turns into a catastrophic failure for the engine.

Even considering that the 80% confidence interval is accurate enough for the purposes of this particular prognosis problem, and that it has been validated using the feature data beyond the 320^{th} cycle of operation, it can be noticed that the predicted upper and lower bounds offer a precise representation for the trend of future feature data for a limited amount of time. In this sense, it is important to note that those bounds are constructed by using the current state estimate for $x_{c,2}(t)$ and that by no means is this model parameter fixed. In fact, the value of the parameter depends on the length of the crack and hence the current RUL estimate is obsolete after a certain period of time, which is exactly what is depicted in Figure 5.7. Thus, there should be a compromise between the desired accuracy in the prediction and the prediction window allowed for early prognosis. There is no clear general solution for the optimal size of the prediction window and it greatly depends on the specifications provided by the customer. For the time being, it is important to note that the prediction window must be large enough to allow corrective actions in the system and avoid catastrophic failures.

5.4. Case Study: UH-60 Planetary Gear Plate. Analysis of Axial Crack Growth

5.4.1. Seeded Fault Test Description and Modeling Aspects

As it has been previously mentioned, the use of particle filtering as a primary tool for state estimation in nonlinear non-Gaussian processes allows to manage the uncertainty inherent to the long term prediction problem. As a consequence, not only it is possible to calculate the expectation of the evolution in time of a failure mode (e.g., a fracture in a rotatory piece of equipment) from a set of measurements (e.g., vibration-based features), but also to establish a consistent methodology for the generation of a probability density function (pdf) associated with the prediction. These capabilities are especially relevant when dealing with processes and phenomena that are not entirely understood, such as the growth of cracks due to fatigue of material. Consider, for instance, the case of prognosis for the evolution of an axial crack on the plate of the UH-60 planetary gearbox; see Figure 3.4 and Figure 5.8.

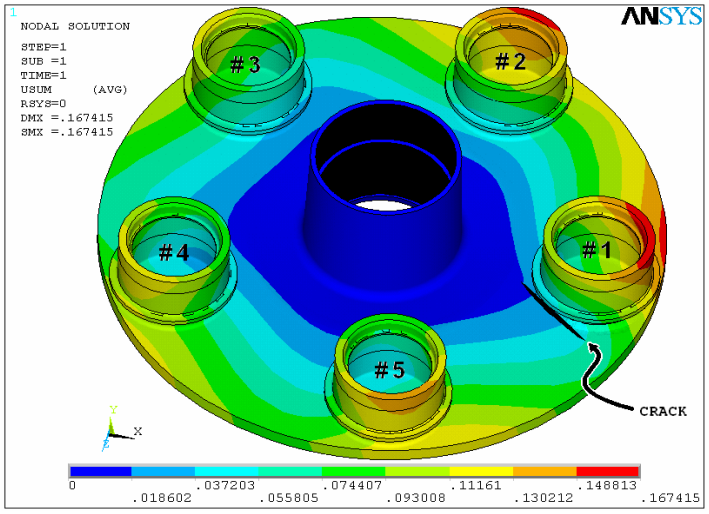

Figure 5.8. ANSYS model of the planetary gear plate, showing crack location.

Although it is well known that this fault mode can lead to a critical failure condition in the aircraft, there was no certain way to determine its existence except by a detailed inspection of this piece of equipment; a very expensive procedure. Under this scenario, the use of algorithms capable of estimating the RUL by only analyzing vibration-based features becomes attractive and helps to decrease operational costs.

With the purpose of testing the feasibility and efficacy of such techniques, a seeded fault test was conducted to collect fault data under a known loading profile. In this test, the crack was artificially grown until it reached a total length of 1.34 inches, after which the gearbox was forced to operate with load changes varying from 20% to 120% of a reference load in a 3 (min) ground-air-ground (GAG) cycle (see Figure 5.9). Given the fact that the initial crack length is known in this case, a deterministic prognosis approach may be considered at first to estimate bounds for the failure time instant.

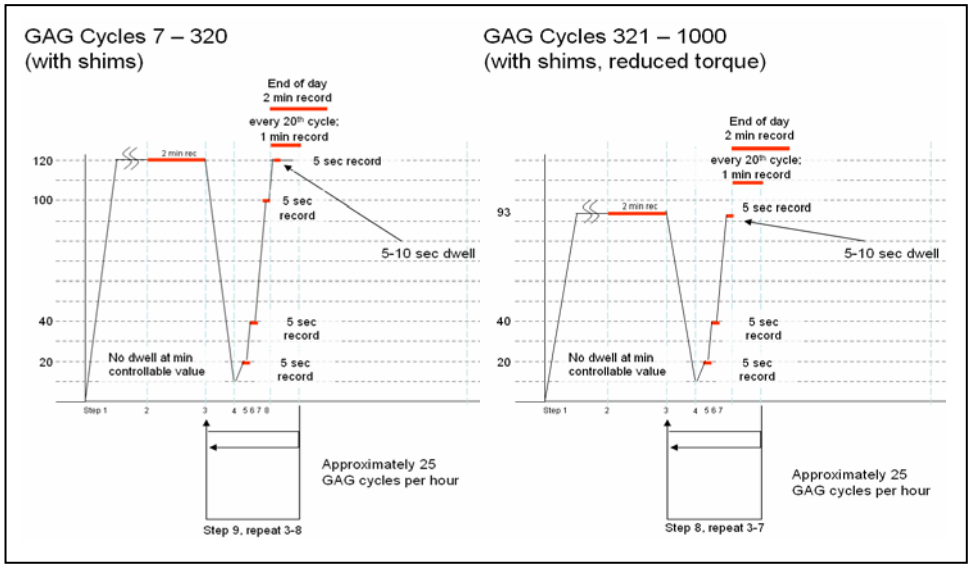

Figure 5.9. Loading profile diagram versus GAG cycles

From material structure theory [35], it is well known that the crack growth evolution may be explained by using an empirical model such as the Paris's Law (5.09), given the proper set of coefficients:

$$\frac{dL}{dn} = C \cdot \left(U(n) \cdot \Delta K(n)\right)^m , \tag{5.09}$$

where L is the total crack length, C and m are material related coefficients, n is the cycle index, $U(n)$ is a parameter that models the effect of crack closure during cycle n and $\Delta K(n)$ is the crack tip stress variation during the cycle n, measured in ($MN/m^{3/2}$). Although simple, model (5.09) requires the computation of two critical parameters to be used in any prognosis routine: $\Delta K(n)$ and $U(n)$.

The stress $K(n)$ may be estimated for a constant load (usually 100%) by using finite element analysis (FEA) tools such as ANSYS for different crack lengths and crack orientation geometries. Considering a proportional relationship between the stress in the tip of the crack and the load percentage, it is in fact possible to construct a mapping relating both the current crack length and load variation per cycle with $\Delta K(n)$.

Although the former piece of information is helpful, it is insufficient for estimating the evolution of the crack length. On the one hand, the closure effect parameter $U(n)$ cannot be efficiently measured and only empirical approximations exist for certain materials, such as Ti-6A1-4V. Even in the case of this particular material, only upper and lower bounds may be computed and therefore it is impossible to compute

expectations and/or determine statistically the validity of confidence intervals. On the other hand, the crack length has to be first estimated to come up with an approximate value for $\Delta K(n)$ and therefore any estimation error will affect the accuracy of the long term prediction.

Keeping in mind all previously mentioned limitations, and using both the known initial condition for the crack length as the starting point and the deterministic model (5.09) in a recursively manner, it is still possible to generate coarse estimates for both the upper and lower bounds for the crack growth evolution [37]; see Figure 5.10.

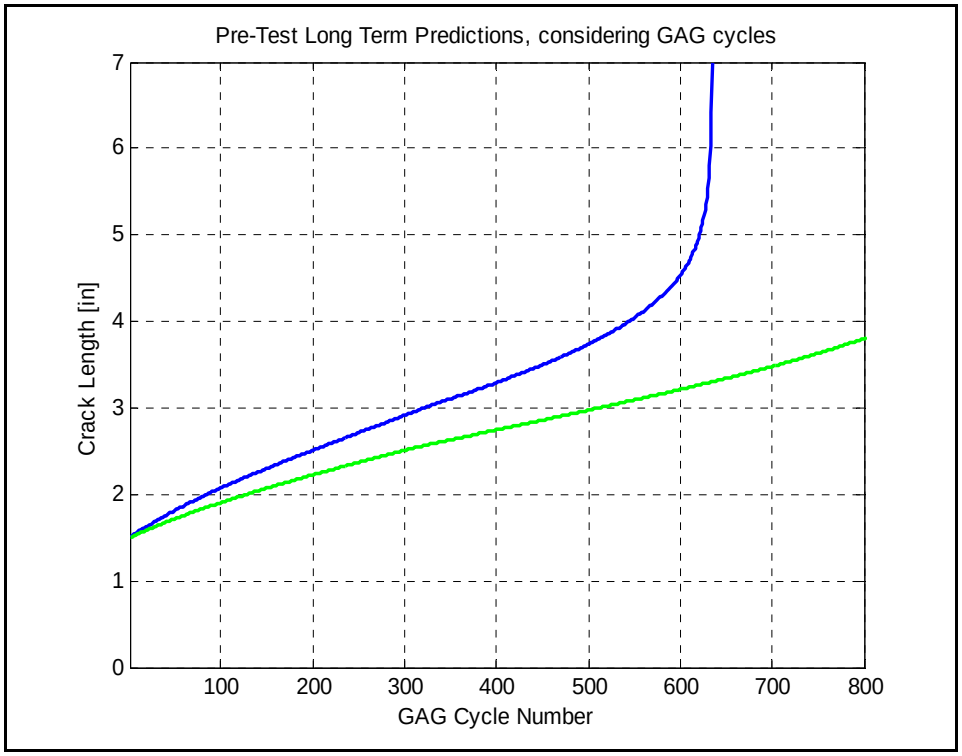

Figure 5.10. Deterministic bounds for crack length evolution vs. GAG cycles. The blue line represents a plain-strain case, while the green line is a plain-stress case. A prognosis procedure based on this approach necessarily has to consider the upper bound to avoid catastrophic failures.

Since a deterministic approach does not provide information about the probability associated with the each of the bounds, a regular maintenance procedure is forced to consider the upper bound to avoid any catastrophic failures. Moreover, it is very difficult to consider changes in the operational conditions and manufacturing differences between aircrafts, and thus the deterministic approach offers little room for on-flight corrective actions. Lastly, it is difficult to determine confidence intervals for the RUL of a particular piece of equipment.

5.4.2. A Particle-Filtering-based Framework for Prognosis of an Axial Crack in a Planetary Carrier Plate

The inclusion of process data, measured and pre-processed in an on-line fashion, tremendously improves what can be achieved in terms of RUL estimation and prognosis, since this inclusion of data provides feedback about the health condition of the process under observation. Indeed, the use of features based on the ratio between the fundamental harmonic and the sidebands in the vibration signal spectrum [37] gives the basis for the implementation of any of the particle-filtering-based prognosis methodologies introduced in Chapter 4. Consequently, under this new approach, not only it is possible to estimate the expected growth of the crack, but also the unknown closure parameter $U(n)$ in the crack growth model (5.09) and the RUL pdf, thus enabling the computation of statistics such as expectations and confidence intervals.

As with any filter-based technique, all proposed prognosis methods require the definition of a process model to incorporate the information present in the feature data.

Therefore, the following crack growth state model (based on Paris's Law) has been implemented for purposes of on-line state and model parameter estimation and RUL pdf estimation by using a particle-filtering-based framework for prognosis:

$$\begin{cases} L(t+1) = L(t) + C \cdot \alpha(t) \cdot \left\{ \left(\Delta K_{inboard}(t) \right)^m + \left(\Delta K_{outboard}(t) \right)^m \right\} + \omega_1(t) \\ \alpha(t+1) = \alpha(t) + \omega_2(t) \\ \Delta K_{inboard}(t) = f_{inboard}\left(\text{Load}(t), L(t) \right) \\ \Delta K_{outboard}(t) = f_{outboard}\left(\text{Load}(t), L(t) \right) \end{cases} \quad , \quad (5.10)$$
$$\text{Feature}(t) = h(L(t)) + v(t)$$

where $L(t)$ is the total crack length estimation at GAG cycle t, $\alpha(t)$ is an unknown time-varying model parameter to be estimated (unitary initial condition), C and m are model constants related to material properties, ΔK is the variation in crack tip stress due to the load profile and the current crack length (estimated through off-line analysis of the system with ANSYS) and $\omega_1(t)$, $\omega_2(t)$ and $v(t)$ are non-Gaussian white noises.

Process model (5.10) needs a noisy estimate of the crack length, based on the value of the feature data point, to be used in on-line applications. This requirement is satisfied via a nonlinear mapping $h(\cdot)$, which is corrected or improved according to the ground truth crack length data that is acquired (at specific and very limited time instants) from strain gages sensors allocated on the surface of the planetary gear plate.

As a result, the proposed scheme considers two update loops running in parallel. The first one, referred to as the *inner loop*, basically uses the feature data and the

previous state pdf estimate to update the crack length and model parameters and thus, the RUL pdf estimate through any of the prognosis approach discussed in Chapter 4. On the other hand a second loop, namely the *outer loop*, revises the nonlinear mapping $h(\cdot)$ between the vibration-based feature value and the crack length every time strain gage data is received. For future on-line applications, it may be assumed that the nonlinear mapping $h(\cdot)$ would still be valid, save for minor adjustments.

A particle-filter-based framework for prognosis based upon model (5.10) provides a state pdf estimate that is updated every GAG cycle. In this pdf estimate, each particle represents a realization of the state vector (namely, both the crack length and the unknown model parameter) that is used as an initial condition for a predicted trajectory, assuming a particular operating regime (e.g., an expected loading profile). The statistical information contained in all predicted trajectories (particularly the evolution in time of the estimated state probability density function) is summarized into a RUL pdf via the definition of hazardous thresholds (a hazard threshold may be understood as a hazard zone with zero variance) for the system under analysis. As a result, at any given time instant, each particle determines: (1) an initial condition for a long term prediction, and (2) a probability associated to that prediction. Figure 5.11 and Figure 5.12 illustrate this fact, by depicting each plausible long term prediction with a different color.

The time instant when each predicted trajectory reaches a given threshold defines a probable failure time and thus, a realization of the RUL probability density function – also referred to as the TTF pdf – for the system under testing (in this particular case the planetary gear plate). The probability associated with this event is the same as the one

linked to the particle that was used as initial condition for the corresponding long term prediction. The collection of all these failure times and their probabilities defines the RUL pdf; once this pdf is estimated, RUL expectations, 95% confidence interval for long term predictions and ± 3 sigma intervals may be also computed. Table 5.1 shows the results for this particular case study, and compares them with the ground truth data that was supplied from strain gages allocated on the surface of the plate.

Ground truth data points (i.e., crack length measurements from strain gages) shown in Table 5.1 were provided, by the personnel at the test facility, incrementally up to 650 GAG cycles in a "*blind*" test format [42]. Thus, for instance, the prediction result of Table 5.1 for GAG #36 (1.60") has been obtained at GAG #0 knowing only the initial crack length. Subsequently, the predicted value for GAG #100 (2.40") has been obtained at GAG #36 after the ground truth data value of 2.00" was used to adjust the nonlinear mapping $h(\cdot)$, as shown in Fig. 5.11. The prediction for GAG #230 was made at GAG #100, and so on so forth.

Every time a new point of ground truth data is included, a more accurate initial condition for the prediction algorithm is estimated, and hence the overall precision of the algorithm is enhanced. The modularity of the proposed approach allows on-line modification of the set of thresholds considered in the analysis, if it is required to increase the hazard level.

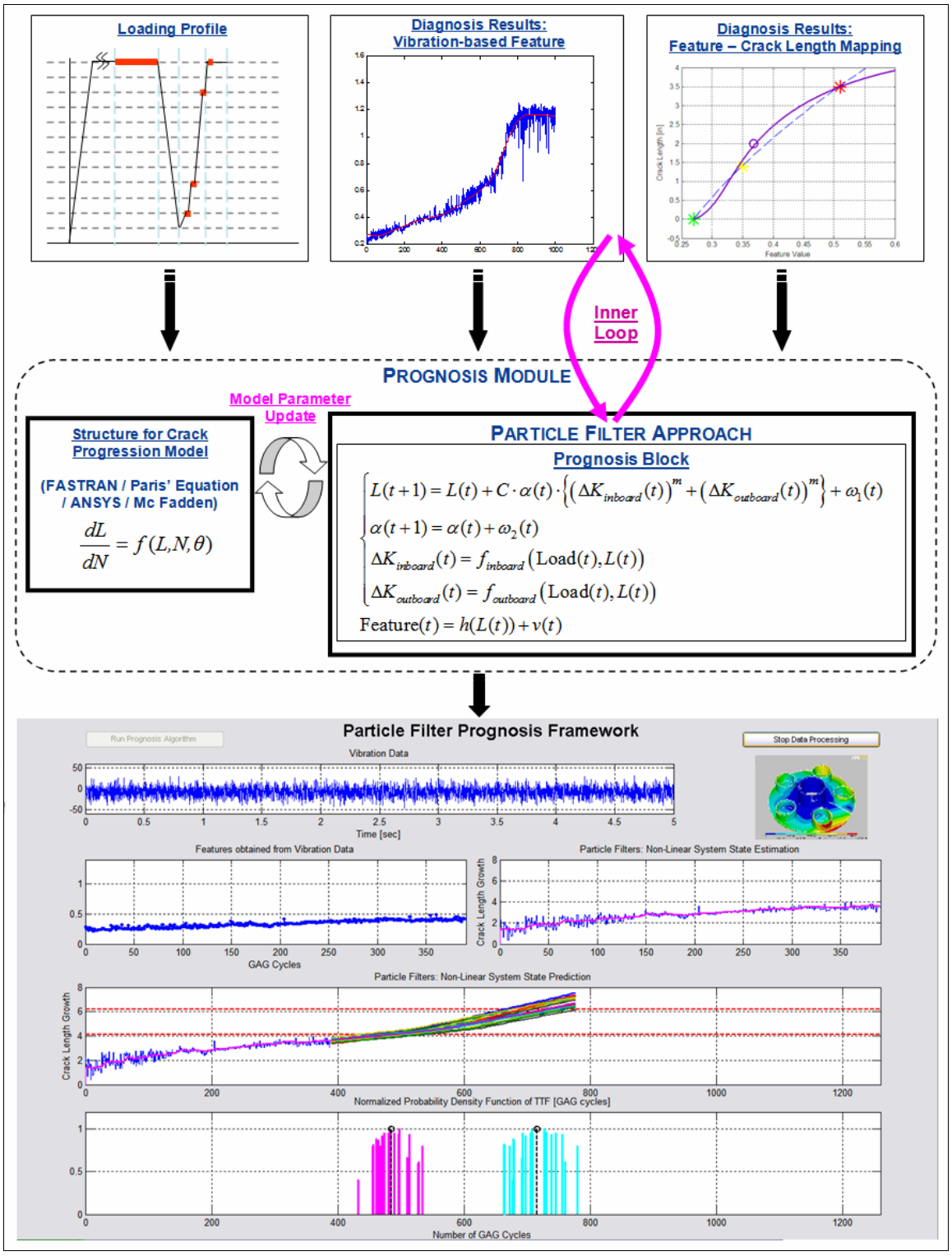

Figure 5.11. Particle-filtering-based approach for prognosis for the study of crack growth in a planetary gear plate. Results for two thresholds are included: the magenta pdf is associated with a threshold of 4" and the cyan pdf with 6.2"

Table 5.1. Prediction results for particle-filtering-based approach for prognosis. Tabled values include expectations, 95% confidence values, and 3σ intervals for the crack length at particular GAG cycles. Table has been completed in a "blind" test manner, i.e. predictions only considered data collected until the previous tabled GAG cycle

Measured Crack Length		Confidence Intervals				
GAG	Crack Length (inches)	–3σ	–95%	Mean	+95%	+3σ
0	1.34	N/A	N/A	1.34	N/A	N/A
36	2.00	0.74	1.03	1.60	2.17	2.46
100	2.50	1.93	2.09	2.40	2.71	2.87
230	3.02	2.73	2.79	2.90	3.01	3.07
400	3.54	3.41	3.54	3.80	4.06	4.19
550	4.07	3.85	4.11	4.30	4.60	4.75
650	4.52	4.20	4.48	4.71	5.08	5.70
750	6.78	6.38	6.42	6.61	6.76	6.84

To illustrate this fact more clearly, consider that the prediction algorithm is launched at GAG cycle 100. Crack length thresholds at 3.0", 3.5" and 4.5" may be established at that time. Given this scenario, the prediction algorithm provides answers to the question: what are the expected (in a probabilistic sense) times at which the crack will reach the corresponding lengths of 3.0", 3.5" and 4.5"? By estimating the RUL pdf, the algorithm supplies the RUL expectation (mean time) and the 95% confidence interval for each case.

As the crack length evolves in time, however, the hazard thresholds can be easily modified to continue the analysis of its growth, eventually reaching the condition of Figure 5.12 where only one remaining hazard threshold is of interest (~6.2") with a TTF expected value of 713 GAG cycles, or equivalently an expected RUL of 325 GAG cycles, which is extremely close to the value of 714 GAG cycles that was provided in the ground

truth data set for the failure time. At this point, it is essential to note that the accuracy of the algorithm has been validated at every step of the "*blind*" test, confirming the robustness of the approach with respect to changes in the load profile (depicted in Figure 5.9) and/or the signal to noise ratio of the feature-based noisy crack length estimation (which steadily improved as the crack length increased, see Figure 5.12).

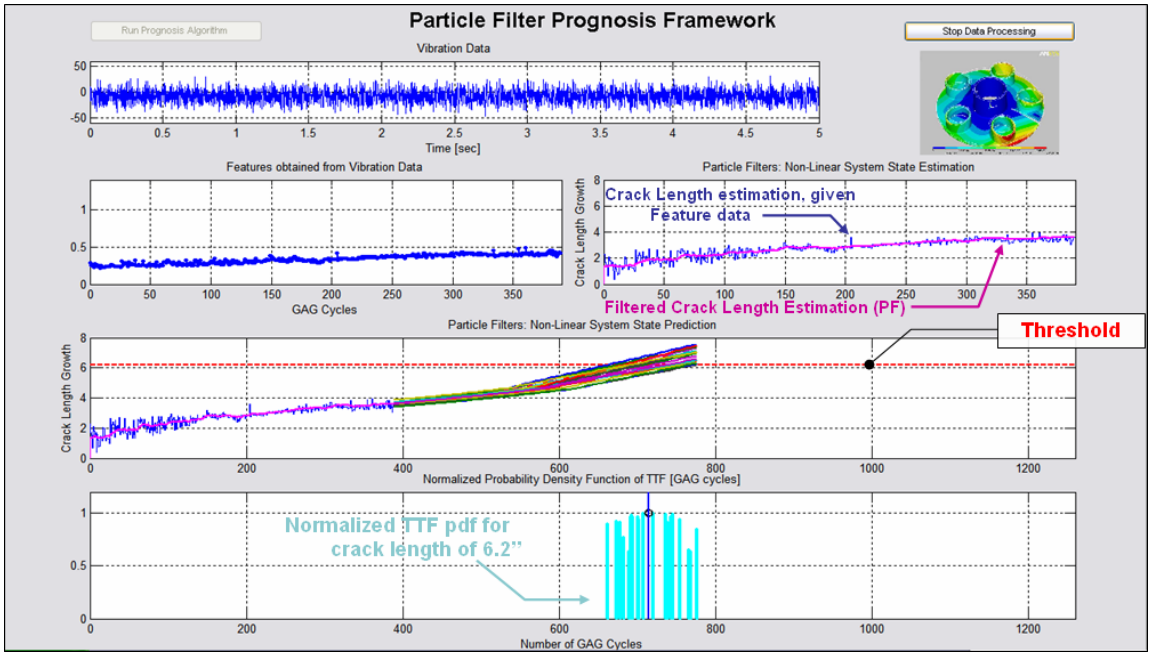

Figure 5.12. Prediction results for a single hazard threshold. The RUL pdf estimate for a threshold of 6.2" has been computed at the 400th GAG cycle, providing a prediction window of 313 GAG cycles (approximately 15.65[hrs])

Given the particle-filtering-based pdf state estimate, additional information about the operating conditions of the system may be also extracted. Consider, for instance, the estimate of the parameter $\alpha(t)$ in model (5.10) that is depicted in Figure 5.13. Sudden changes in the estimate of the unknown model parameter are clear indicators of a change in the testing operating conditions, as in the case depicted in Figure 5.13 where the

maximum value of the load applied to the carrier plate was decreased approximately at GAG cycle #320.

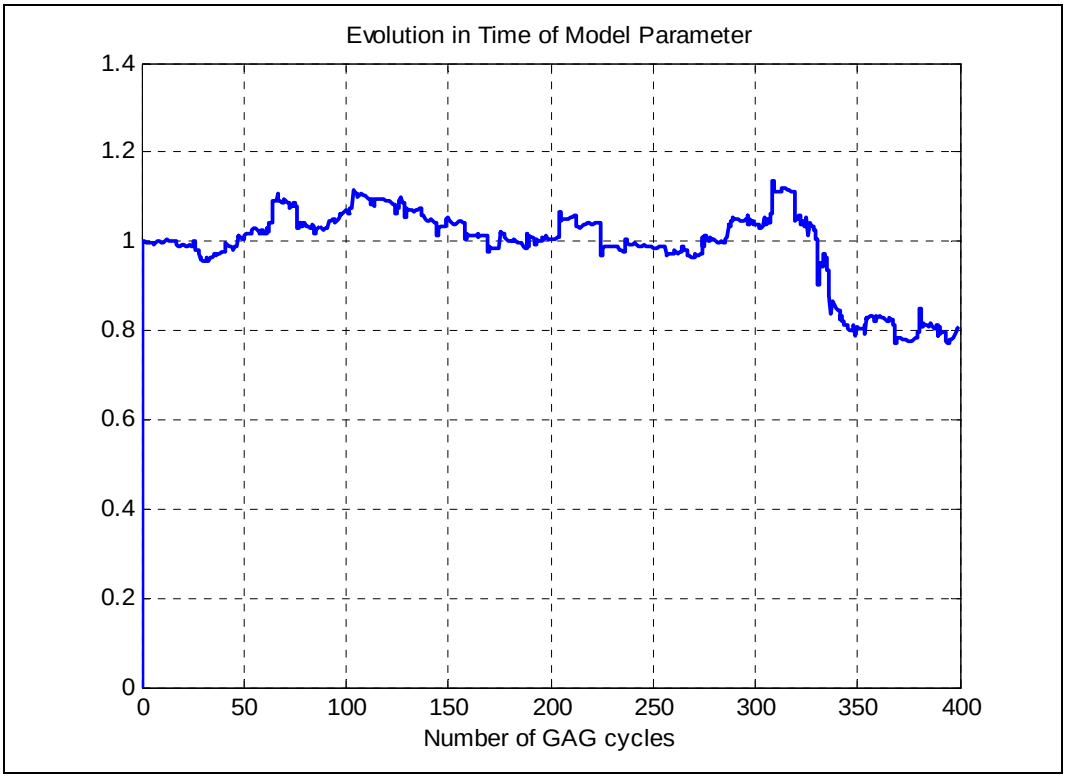

Figure 5.13. Time-varying model parameter vs. GAG cycles. The sudden drop in the estimate of the model parameter at GAG cycle #320 indicates a change in the operating conditions of the seeded fault test. In fact, this change corresponds to a decrement in the maximum value of the load profile applied to the carrier plate

When only one particular threshold is of interest, it is always possible to use historical failure data or customer specifications to construct a hazard zone, as in Figure 5.14 where an average length of 6.2" has been considered to generate the RUL pdf estimate, and hence, its expectation. The 95% confidence interval generated under these conditions has been successfully validated (see vertical blue line across the non-Gaussian RUL pdf in Figure 5.14, representing the ground truth failure data point) during the

"*blind*" test, and thus the prognosis approach in general has proven to meet all necessary requirements to be considered a satisfactory solution for the crack growth problem.

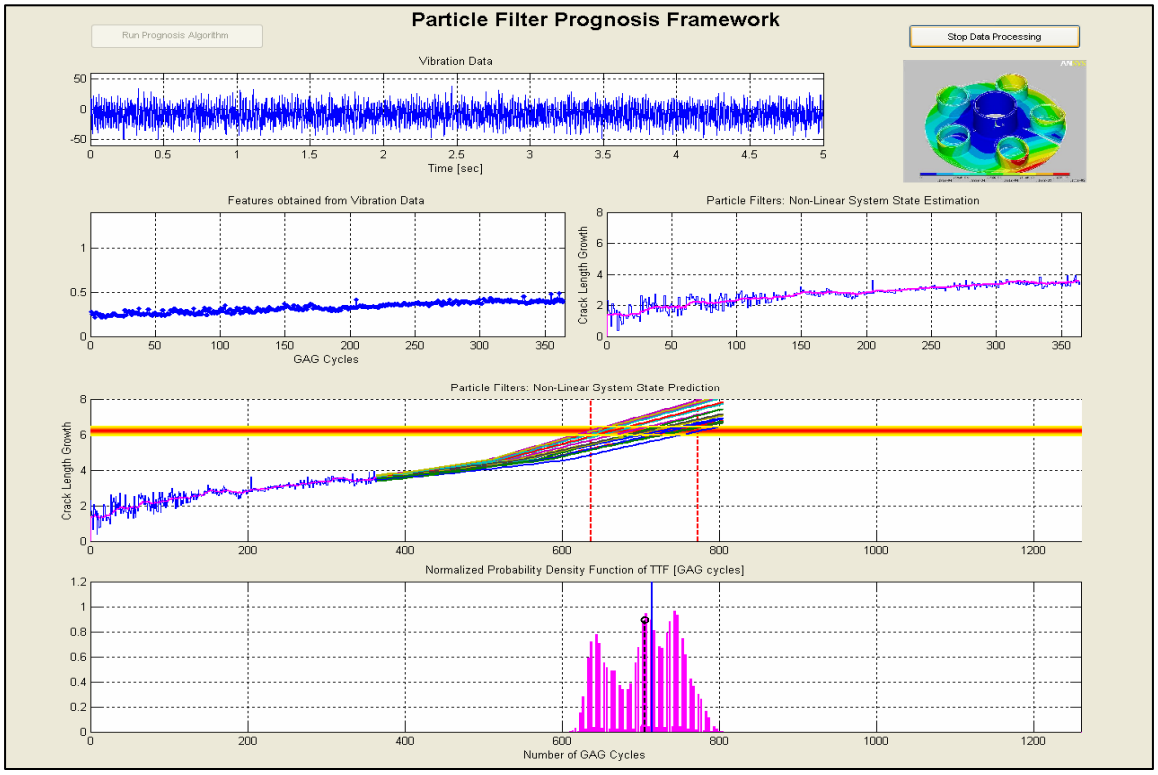

Figure 5.14. Prognosis results for a unique hazard zone at 6.2". (a) Vibration data over a time window of 5[sec]. (b) Vibration-based feature data vs. GAG cycles. (c) Noisy and filtered estimates for the crack length. (d) RUL pdf estimate for a hazard zone around 6.2", the vertical blue line corresponds to ground truth data

It must also be noticed that the prognosis technique looses accuracy when analyzing the growth of crack lengths exceeding 6.75" (and thus, when one of the tips of the crack is about to exit from the plate surface), since the FEA model developed for the planetary gear plate does not describe the stresses that affect the tips of the crack under those conditions. Regardless of the aforementioned fact, the proposed approach has

proven to be efficient, accurate and precise to solve the prognosis problem for any crack length within the range of interest (crack lengths smaller than 6.2").

On another topic, a few words must be said about the concepts of "correctness" and "safety" of the prognosis results obtained from the proposed particle-filtering-based framework. "Correctness" may be mathematically defined in terms of the generation of confidence intervals for the RUL of the system, which in turn are constructed from the pdf estimates that the prognosis framework provides: a prognosis result is "correct" if the ground truth failure data fall within the predicted confidence interval.

"Safety", on the other hand, is a more relaxed definition. A prognosis result could be considered "safe" as long as the *infimum* of the predicted confidence interval is smaller than or equal to the ground truth failure time. For example, consider Figure 5.15, where the evolution in time of the 95% confidence intervals for the RUL of the carrier plate is depicted, considering a hazard zone around 4.5". Although the ground truth failure time (blue horizontal line) sometimes falls outside the 95% confidence interval, the prognosis result is always "safe" in the sense that the system could not possibly fail before the lower bound of the provided time interval.

From the "correctness" standpoint, though, it may seem that the algorithm does not always provide a correct prognosis. However, it must be noted that the ground truth data correspond to the failure time *after* the operating conditions of the system changed (GAG #320). If the maximum load were not decreased (from 120% to 93%), the crack

would have grown faster, and therefore the 95% confidence interval provided at the GAG #320 (see Figure 5.15) may have been perfectly "correct".

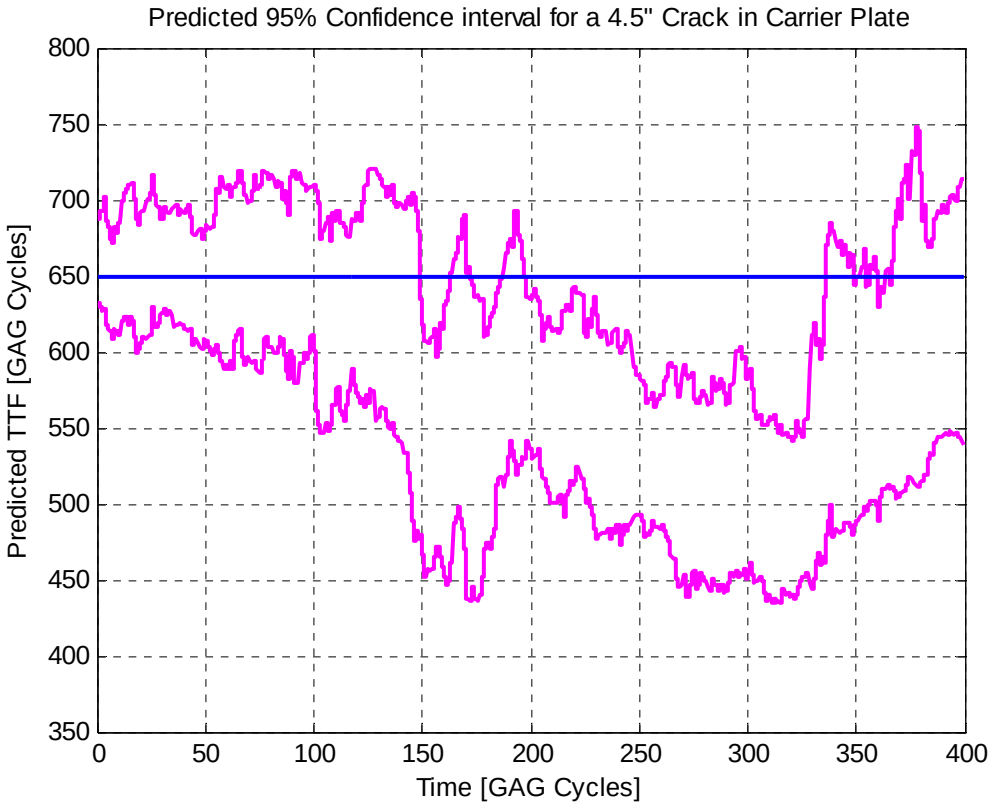

Figure 5.15. Evolution in time of 95% confidence intervals for the RUL, considering a hazard zone around 4.5". The blue horizontal line indicates the ground truth failure data.

It is clear from Figure 5.15 that the prognosis framework adapts its output after the change in the loading profile. Final prognosis results for this hazard zone are shown in Figure 5.16.

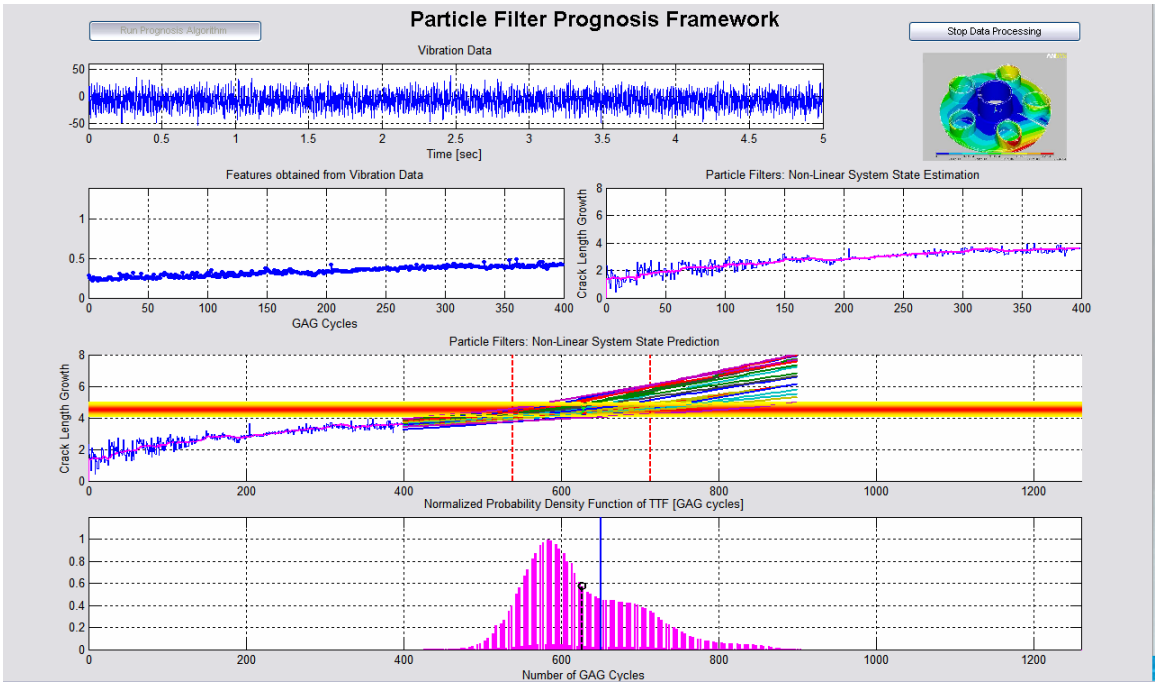

Figure 5.16. Prognosis results for a unique hazard zone at 4.5". (a) Vibration data over a time window of 5[sec]. (b) Vibration-based feature data vs. GAG cycles. (c) Noisy and filtered estimates for the crack length. (d) RUL pdf estimate for a hazard zone around 4.5", the vertical blue line corresponds to ground truth data

Finally, it is important to mention that the author has thoroughly compared the proposed methodology with respect to an EKF-based approach for long term predictions. Results are always favorable for the proposed particle-filtering-based prognosis scheme in terms of accuracy and precision of the RUL pdf estimate; as was shown in the results of Section 5.2.

Considering all of the above, it is possible to affirm that the proposed methodology offers, in this case, a complete and modular solution to the prognosis problem, which has been tested with excellent results and validated at several stages within the progression of the seeded fault.

Furthermore, the particle filtering framework for the prediction of the RUL may be easily implemented in real time on-board a HUMS or other health monitoring platform for on-line applications; in fact, an integrated architecture that combines vibration data processing, feature extraction, fault diagnosis and failure prognosis based on this concept is described in [36], [38].

The excellent quality of the obtained results not only validates the proposed methodology, but also gives support for the implementation of more sophisticated techniques such as APF or RPF and noise structure adaptation techniques to improve both the on-line state and RUL pdf estimates.

5.4.3. A Graphical User Interface for On-Line Analysis

A graphical user interface (GUI) has been designed to display the results of the integrated algorithms for diagnosing and prognosticating the carrier plate crack; see Figure 5.17. The GUI conveys system-health information in real-time, including sensor data validation, extracted features for different sensors and engine torque levels, and 95% confidence intervals for the crack length and time-to-failure (TTF) of three length thresholds used in lieu of hazard zones. The GUI allows an operator to specify the threshold values and the probabilities of false alarms and detection [42].

Early detection, and precision and accuracy of the prognosis results were evaluated. The baseline pdf for each feature was constructed considering vibration data from healthy systems (no crack). The loading profile of the system was a controlled

variable. The GUI was run after the seeded fault test was completed, but real-time operation was simulated. The robustness of the GUI was also evaluated against missing data points (e.g., problems in data acquisition).

In general, model parameters can be determined in either an off-line or on-line fashion. The off-line approach determines the parameter values before the health monitoring architecture is run onboard an aircraft. The pre-determined parameters in the present implementation include maps relating vibration feature values to crack lengths (diagnosis) and values of ΔK for different amounts of damage (prognosis). These values are stored in a database to be accessed as needed by the monitoring architecture. The present architecture showed that all necessary computations can be performed in real time in most microprocessor currently available.

Results proved the efficacy of the algorithms. The crack is detected after just 13 load cycles out of a total run time of more than 1000. Post-test ground-truth crack lengths are always within the 95% confidence intervals of the prediction. Results also demonstrate that it would be feasible to implement the architecture on-board a helicopter. In addition, the combination of model-based and data-driven techniques provided two important advantages. First, the prediction results were very robust, since the effect of outliers in the data was mitigated by the model. Second, the architecture allows on-line adaptation whenever the system undergoes environment changes, e.g., changes in the loading profile [42].

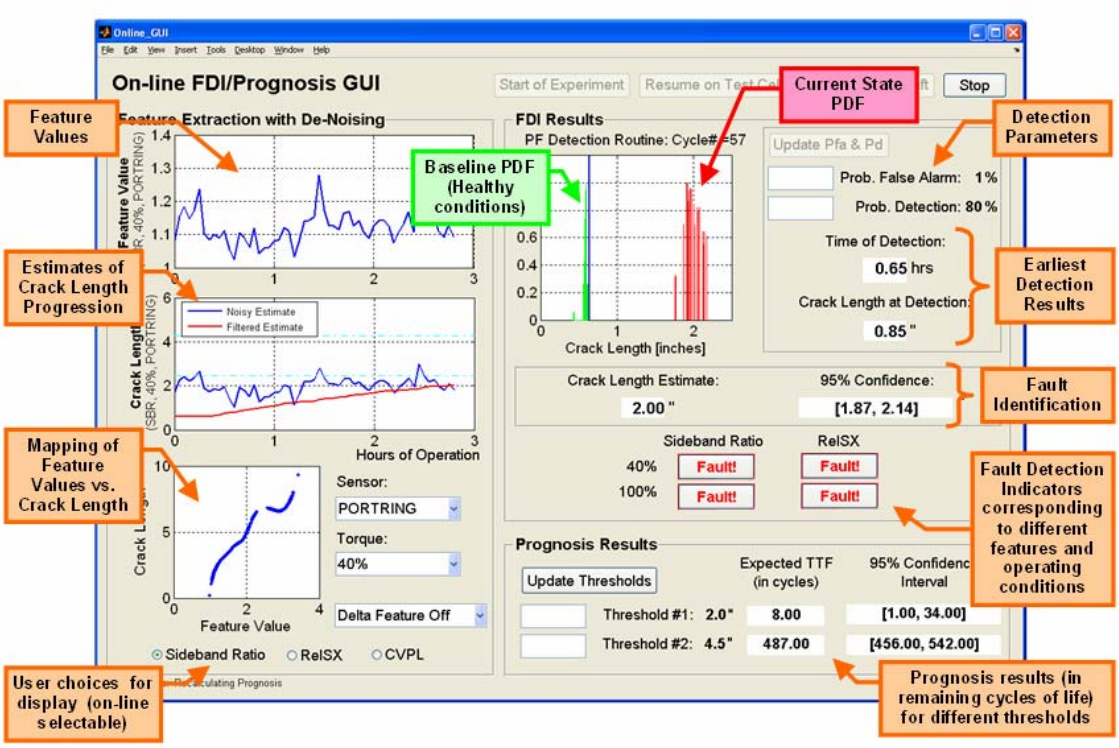

Figure 5.17. Graphical user interface (GUI) displaying results from both diagnostic and prognostic routines. All implemented methodologies considered a particle-filtering-based framework. Picture extracted from [42]

6. CONCLUSIONS AND SUGGESTED FUTURE WORK

This thesis presents the theoretical foundation, implementation, testing and assessment of an on-line particle-filtering-based framework for fault diagnosis and failure prognosis. Several approaches have been offered as a means of implementing this framework, all of them based on the fact that the current state pdf estimate may be used to determine the operating condition of the system and predict the progression of a fault indicator, given a dynamic state model and a set of process measurements. In all of these approaches, the task of estimating the current value of the fault indicator, as well as other important changing parameters in the environment, involves two basic steps: the prediction step, based on the process model, and an update step, which incorporates the new measurement into the a priori state estimate.

The general diagnosis framework introduced in this thesis has been successful and efficient in pinpointing abnormal conditions in the operation of a nonlinear system. This framework provides an estimate of the probability of a fault for a set of known fault modes that is updated in real time, using a hybrid nonlinear state-space model for the system, a characterization of the plant behavior under nominal operational conditions (baseline data), and a set of measurements. In addition, this framework allows the means to perform swift transitions between FDI and prognostic-oriented applications. The description of the proposed methodology includes a step-by-step procedure that allows the computation of *type I* and *type II* detection errors, and also the execution of classical statistical hypothesis tests for fault detection. In addition, this methodology also provides

a means for detecting the existence of unknown anomalies in the monitored system, with specific confidence levels (concept of anomaly detector).

A novel particle-filtering-based framework for on-line failure prognosis has also been presented in this thesis. This framework generates an estimate of the probability of failure at future time instants (RUL pdf) in real time. Information about time-to-failure (TTF) expectations, statistical confidence intervals, and other hypothesis tests is also provided by combining state pdf estimates, long-term predictions, and empirical knowledge about critical conditions for the system (also referred to as the hazard zones). Several approaches for the propagation of the state pdf in time are discussed and, in particular, it is shown that a combination of resampling schemes in long-term predictions and Epanechnikov kernels helps to reduce the impact of model errors, and simultaneously offers a balanced answer in terms of accuracy and precision in RUL estimates. In addition, it is shown that an approach based solely on the expectation of the long-term prediction also provides acceptable results, and moreover, it is suitable for on-line applications with limited computational resources. Two successful case studies are presented to validate the performance of the proposed methodology with real failure data, both of them using a simple SIR particle filter implementation and an expectation-based method for long term prediction generation. Both studies provide insight about how model inaccuracies and/or customer specifications (hazard zone definition or desired prediction window) may affect the algorithm performance.

This thesis also provides insight about the concept of *outer correction loops* to update parameters of significance in the overall performance of FDI and/or prognostic

algorithms. In particular, two *correction loops* are described and discussed in detail: (1) an autoregressive correction algorithm utilized to improve accuracy in RUL expectations, and (2) a model parameter update procedure that facilitates identification of nonlinear systems undergoing changes in operational conditions. Both of them offer good results and illustrate how the accuracy of the prognostic algorithm may be significantly enhanced when several learning loops – combining model-based and data driven techniques – are working in parallel.

Several publications have been generated as part of the research work hereby discussed. On the one hand, results presented in the case study of detection and analysis of cracks in the blades of a turbine HPC disk (Sections 3.2 and 5.3) have been disclosed to the public in the following publications:

- Orchard, M., Wu, B. and Vachtsevanos, G., "A Particle Filter Framework for Failure Prognosis," Proceedings of WTC2005, World Tribology Congress III, Washington D.C., USA, 2005.

- Orchard, M. E. and Vachtsevanos, G., "A Particle Filtering-based Framework for Real-time Fault Diagnosis and Failure Prognosis in a Turbine Engine," 15th Mediterranean Conference on Control and Automation MED'07, Athens, Greece, July 2007.

On the other hand, results obtained from the analysis of the detection and prognosis problem in the case of an axial crack in a UH-60 planetary gear carrier plate have been summarized in the following publications:

- Patrick R., Orchard, M., Zhang, B., Koelemay, M., Kacprzynski, G., Ferri, A., Vachtsevanos, G., "An Integrated Approach to Helicopter Planetary Gear Fault Diagnosis and Failure Prognosis," 42nd annual Systems Readiness Technology Conference, AUTOTESTCON 2007, Baltimore, USA, September 2007.

- Zhang, B., Khawaja, T., Patrick R., Vachtsevanos, G., Orchard, M., Saxena, A., "Use of Blind Deconvolution De-Noising Scheme in Failure Prognosis," 42nd annual Systems Readiness Technology Conference, AUTOTESTCON 2007, Baltimore, USA, September 2007.

- Marcos E. Orchard and George J. Vachtsevanos, "A Particle Filtering Approach for On-Line Fault Diagnosis and Failure Prognosis". To be published in a special issue of Transactions of the Institute of Measurement & Control on Intelligent Fault Diagnosis & Prognosis for Engineering Systems.

- Bin Zhang, Taimoor Khawaja, Romano Patrick, George Vachtsevanos, Marcos Orchard, Abhinav Saxena, "A Novel Blind Deconvolution De-

Noising Scheme in Failure Prognosis". Submitted to Transactions of the Institute of Measurement and Control.

In addition, results obtained from the application of a particle-filter-based framework for FDI and failure prognosis in other types of dynamic systems have generated the following publications:

- Abbas, M., Ferri, A., Orchard, M., Vachtsevanos, G., "An Intelligent Diagnostic/Prognostic Framework for Automotive Electrical System," 2007 IEEE Intelligent Vehicles Symposium-IV'07, Istanbul, Turkey, June 13-15, 2007.

- Bin Zhang, Marcos E. Orchard, Abhinav Saxena, and George J. Vachtsevanos, "Rolling Element Bearing Feature Extraction and Anomaly Detection Based on Vibration Monitoring". Submitted to the 2008 American Control Conference.

Future work suggested from this research includes: (1) an in-depth study of the performance of the proposed particle-filtering-based diagnosis framework in the presence of a time-varying baseline pdf, (2) a detailed analysis of feedback stability and convergence of *outer correction loops* in a particle-filter-based framework for prognosis, and (3) the development, testing and impact evaluation of block sampling strategies on the overall performance of the a particle-filtering-based FDI and prognosis framework.

REFERENCES

[1] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, February 2002, pp. 174 – 188.

[2] Doucet, A., de Freitas, N., and Gordon, N., "An introduction to Sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[3] Haug, A. J., "A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian Processes," MITRE Technical Report. MTR 05W0000004, The MITRE Corporation, 2005.

[4] Doucet, A., Godsill, S., and Andrieu, C., "On Sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no 3, pp. 197-208, July 2000.

[5] Andrieu, C., Doucet, A., and Punskaya, E., "Sequential Monte Carlo Methods for Optimal Filtering," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[6] Casella, G., and Roberts, C.P., "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83 (1), pp. 81 – 94, 1996.

[7] Chen, R., and Liu, J. S., "Predictive updating methods with application to Bayesian classification," *Journal of the Royal Statistical Society* (Ser. B), vol. 58, pp. 397 – 415, 1996.

[8] De Freitas, N., "Rao-Blackwellised Particle Filtering for Fault Diagnosis," *IEEE Aerospace Conference Proceedings* (Cat. No. 02TH8593), pt. 4, pp. 1767 – 1772, 2002.

[9] Kong, A., Liu, J. S., and Wong, W. H., "Sequential imputations and Bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, pp. 278 – 288, 1994.

[10] Verma, V., Gordon, G., Simmons, R., and Thrun, S., "Particle Filters for Rover Fault Diagnosis," *IEEE Robotics & Automation Magazine*, pp. 56 – 64, June 2004.

[11] Doucet, A., "On sequential Monte Carlo methods for Bayesian Filtering," Technical Report, Engineering Department, Univ. Cambridge, UK, 1998.

[12] Liu, J. S., "Metropolized independent sampling with comparison to rejection sampling and importance sampling," *Statistics and Computing*, vol. 6, pp. 113 – 119, 1996.

[13] Pitt, M. K., and Shephard, N., "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, pp. 590 – 599, 1999.

[14] Berzuini, C., Best, N., Gilks, W., and Larizza, C., "Dynamic conditional independence models and Markov Chain Monte Carlo methods," *Journal of the American Statistical Association*, vol. 92, pp. 1403 – 1412, 1997.

[15] Crisan, D., "Particle Filters − A Theoretical Perspective," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[16] Gordon, N. J., Salmond, D. J., and Smith, A. F. M., "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, no. 2, pp. 107 – 113, 1993.

[17] Liu, J. S., and Chen, R., "Sequential Monte Carlo methods for dynamic systems," *Journal of the American Statistical Association*, vol. 93, pp. 1032 - 1044, 1998.

[18] Liu, J. S., Chen, R., and Wong, W. H., "Rejection control and sequential importance sampling," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1022 – 1031, 1998.

[19] Kitagawa, G., "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1-25, 1996.

[20] Van der Merwe, R., Doucet, A., de Freitas, N., and Wan, E., "The Unscented Particle Filter," Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, 2006.

[21] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., Eds. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, Suffolk, 1996.

[22] Godsill, S., and Clapp, T., "Improvement Strategies for Monte Carlo Particle Filters," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[23] Zaritskii, V. S., Svetnik, V. B., and Shimelevic, L. I., "Monte Carlo technique in problems of optimal data processing," *Automation and Remote Control*, vol. 12, pp. 95 – 103, 1975.

[24] Musso, C., Oudjane, N., and Le Gland, F., "Improving regularised particle filters," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[25] Liu, J., and West, M., "Combined Parameter and State Estimation in Simulation-Based Filtering," in *Sequential Monte Carlo Methods in Practice*, Doucet, A., de Freitas, N., and Gordon, N., Eds. New York: Springer-Verlag, 2001.

[26] West, M., "Approximating posterior distributions by mixtures," *Journal of Royal Statistical Society* (Ser. B), vol. 48, pp. 70 – 78, 1993.

[27] West, M., "Mixture models, Monte Carlo, Bayesian updating and dynamic models," in *Computing Science and Statistics: Proceedings of the 24th Symposium on the Interface*, Newton, J.H., Ed. Interface Foundation of North America, Fairfax Station, Virginia, pp. 325 – 333, 1993.

[28] Kadirkamanathan, V., Li, P., Jaward, M. H., and Fabri, S. G., "Particle filtering-based fault detection in non-linear stochastic systems," *International Journal of Systems Science*, vol. 33, no. 4, pp. 259 – 265, 2002.

[29] Li, P., and Kadirkamanathan, V., "Particle Filtering Based Likelihood Ratio Approach to Fault Diagnosis in Nonlinear Stochastic Systems," *IEEE Transactions on Systems, Man, and Cybernetics* ─ Part C: Applications and Reviews, vol. 31, no. 3, 2001.

[30] Koutsoukos, X., Kurien, J., and Zhao, F., "Monitoring and Diagnosis of Hybrid Systems Using Particle Filtering Models," International Symposium on Mathematical Theory of Networks and Systems, 2002.

[31] Thrun, S., Langford, J., and Verma, V., "Risk Sensitive Particle Filters," Neural Information Processing Systems (NIPS), December 2001.

[32] Verma, V., Thrun, S., and Simmons, R., "Variable Resolution Particle Filter," Proceedings of the International Joint Conference of Artificial Intelligence, 2003.

[33] Gustafsson, F., and Hriljac, P., "Particle Filters for System Identification with Application to Chaos Prediction," 13th IFAC Symposium on System Identification, Rotterdam, The Netherlands, 2003.

[34] Huang, W., and Dietrich, D., "An Alternative Degradation Reliability Modeling Approach Using Maximum Likelihood Estimation," *IEEE Transactions on Reliability*, vol. 54, no. 2, pp. 310 – 317, 2005.

[35] Ray, A., and Tangirala, S., "Stochastic Modeling of Fatigue Crack Dynamics for On-Line Failure Prognosis," *IEEE Transactions on Control Systems Technology*, vol. 4, no. 4, pp. 443 – 451, 1996.

[36] Orchard, M. E. and Vachtsevanos, G., "A Particle Filtering-based Framework for Real-time Fault Diagnosis and Failure Prognosis in a Turbine Engine," 15th Mediterranean Conference on Control and Automation MED'07, Athens, Greece, July 2007.

[37] Patrick-Aldaco, R., "A Model Based Framework for Fault Diagnosis and Prognosis of Dynamical Systems with an Application to Helicopter Transmissions," Ph.D. Thesis. Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, 2007.

[38] Orchard, M., Wu, B. and Vachtsevanos, G., "A Particle Filter Framework for Failure Prognosis," Proceedings of WTC2005, World Tribology Congress III, Washington D.C., USA, 2005.

[39] Vachtsevanos, G., Lewis, F. L., Roemer, M. J., Hess A., and Wu, B., *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*, Hoboken, NJ, John Wiley and Sons, 2006.

[40] Worden, K., Sohn, H., and Farrar, C. R., "Novelty Detection in a Changing Environment: Regression and Interpolation Approaches," *Journal of Sound and Vibration*, vol. 258(4), pp. 741 – 761, 2002.

[41] Ross, S.M., *Introduction to Probability Models*, Fourth Edition. Academic Press, San Diego, CA, 1989.

[42] Patrick R., Orchard, M., Zhang, B., Koelemay, M., Kacprzynski, G., Ferri, A., Vachtsevanos, G., "An Integrated Approach to Helicopter Planetary Gear Fault Diagnosis and Failure Prognosis," 42nd annual Systems Readiness Technology Conference, AUTOTESTCON 2007, Baltimore, USA, September 2007.